

# Proiect SGBD: Spațiile comerciale din România

## 1. Preambul

---

### 1.1 Alegerea temei

După apariția în anul 1995 a primului magazin alimentar modern de mare suprafață din România, piața vânzărilor cu amănuntul a prins avânt pe tot cuprins țării noastre, dezvoltându-se temeinic chiar și în perioadele de criză (cea economică din 2008 și cea pandemică din 2020). Cifra de afaceri al celui mai mare business din domeniu, Kaufland, a atins în anul 2020 un nou nivel record de 13 miliarde de lei, datorat în primul rând de extinderea numărului de magazine<sup>1</sup>. Această extindere accelerată se înregistrează și în rândul celorlalte companii concurente, astfel că un viitor factor de o deosebită importanță pentru comercianți îl va reprezenta cunoașterea pieței de desfacere de pe teritoriul României, fapt prin care se evidențiază și necesitatea temei abordate pe parcursul acest proiect.

### 1.2 Descrierea aplicației

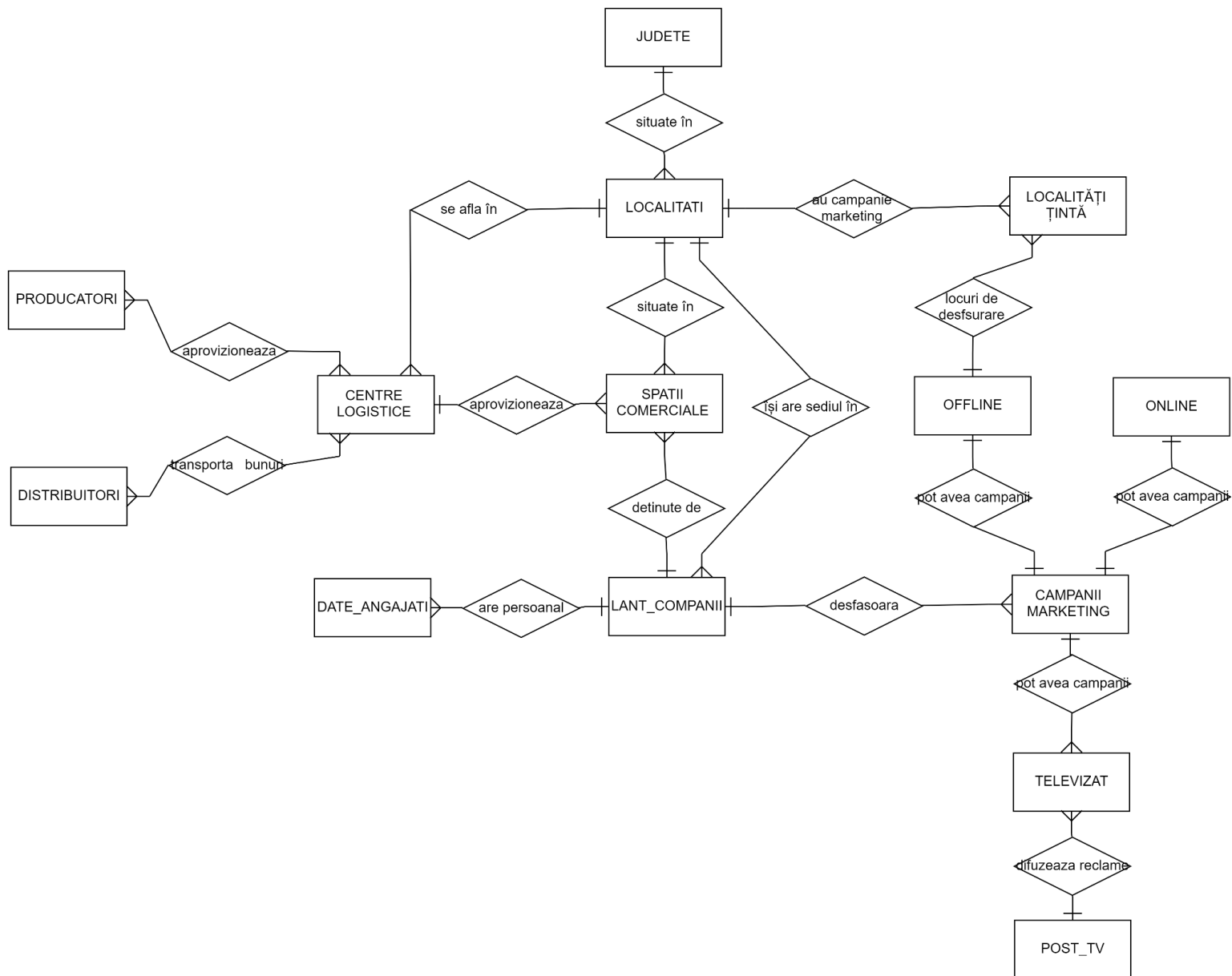
Sistemul de gestiune al magazinelor are ca punct de pornire spațiul comercial, care va fi categorisit în funcție de compania care îl gestionează și despre care se vor cunoaște informații generale privind localizarea (localitatea și județul) și angajații (date salariale și numărul de angajați pentru fiecare job în parte). Pentru ca fiecare magazin să funcționeze corespunzător și ca aprovizionarea să se realizeze facil, aceste vor depinde de un singur centru logistic care să le distribuie marfa. La rândul lor, centrele logistice vor procura produsele de la diverse companii parteneri, ce își pot avea sediul în țară sau în străinătate și care se află în relație directă cu distribuitorii pentru a livra marfa. Pentru a se menține consumatorii la curent cu cele mai noi oferte puse la dispoziție, companiile desfășoară campanii de marketing, care sunt împărțite în funcție de calea de comunicare prin care se publică: offline (reviste, pliante, panouri publicitare), televizate și online. Mediatizarea televizată se va face cu anumite costuri, în funcție de postul de televiziune ales (și în funcție de unul dintre intervalele orare alese: intervalul I de la 08:00 la 16:00 sau intervalul II de la 16:00 la 23:00), în timp ce pentru mediatizarea offline se va ține cont de localitățile țintă.

---

1

[https://www.economica.net/crestere-de-peste-8-a-business-ului-kaufland-in-pandemie-2021-vine-cu-investitii-mai-mari-cu-25\\_503870.html](https://www.economica.net/crestere-de-peste-8-a-business-ului-kaufland-in-pandemie-2021-vine-cu-investitii-mai-mari-cu-25_503870.html)

## 2. Diagrama ERD



### 2.1 Descrierea entităților

**SPATII\_COMERCIALE (#COD\_MAGAZIN)** - entitate ce va conține detalii specifice fiecărui spațiu comercial de pe teritoriul României;

**LOCALITATI (# COD\_LOCALITATE)** - entitate în care se vor regăsi detalii despre toate localitățile în care există cel puțin un magazin, un sediu de companie sau un centru logistic;

**JUDETE (#COD\_JUDET)** - entitate cu informații despre fiecare județ din România;

**LANT\_COMPANII (# COD\_COMPANIE)** - entitate care a ca scop clasificarea spațiilor comerciale în funcție de lanțul comercial din care fac parte (spre ex: Kaufland, Carrefour, Penny etc.);

**CAMPANII\_MARKETING (#COD\_MARKETING)** - entitate care reține detaliile generale despre campaniile de marketing;

**ONLINE (#COD\_ONLINE)** - entitate cu informații despre modalitatea de desfășurare a unei campanii de marketing în mediul online;

**TELEVIZAT (#COD\_TELEVIZAT)** - entitate în care se stochează pentru fiecare campanie de marketing informații despre contractele cu televiziunile partenere;

**POST\_TV (# COD\_TV)** - entitate în care se regăsesc date despre posturile de televiziune;

**OFFLINE (# COD\_OFFLINE)** - entitate cu informații despre modalitatea de desfășurare a unei campanii de marketing în mediul offline;

**LOCALITATI\_TINTA (# COD\_TINTA)** - entitate în care se regăsesc id-urile tuturor localităților aflate în aria de interes a unei campanii de promovare offline;

**CENTRE\_LOGISTICE (#COD\_LOGISTIC)** - entitate în care se regăsesc detalii despre centrele logistice ale lanțurilor de magazine;

**PRODUCATORI (#COD\_PRODUCATOR)** - entitate care ține informații despre producătorii care aprovizionează centrele logistice;

**DISTRIBUITOR (#COD\_DISTRIBUITOR)** - entitate care menține detalii despre cei care preiau marfa de la producători și o transportă spre centrele logistice;

**DATE\_ANGAJATI (# COD\_ANGAJATI )** - entitate care conține detalii despre angajații lanțurilor de magazine.

## 2.2 Descrierea relațiilor

**LOCALITATI - SPATII\_COMERCIALE:** one to many, o localitate poate găzdui mai multe centre comerciale diferite;

**JUDETE - LOCALITATI:** one and only one to many, fiecare localitate se poate regăsi doar într-un județ (fiecare județ conține mai multe localități diferite);

**LANT\_COMPANII – SPATII\_COMERCIALE:** one and only one to many, fiecare magazin poate fi parte a numai unui singur lanț comercial;

**LANT\_COMPANII – LOCALITATI:** one to one, fiecare companie își poate avea sediul central într-o singură localitate;

**LANT\_COMPANII – DATE\_ANGAJATI:** one and only one to many, fiecare companie poate avea mai multe tipuri de angajați pentru care se stochează date;

**LANT\_COMPANII – CAMPANII\_MARKETING:** one to one or many, fiecare lanț de magazin poate desfășura una sau mai multe campanii de marketing;

**LANT\_COMPANII – CENTRE\_LOGISTICE:** one to many, fiecare companie deține mai multe centre logistice;

**CAMPANII\_MARKETING – TELEVIZAT:** one to many, fiecare campanie de marketing poate avea una sau mai multe acțiuni televizate de promovare;

**POST\_TV – TELEVIZAT:** one to many, fiecare post de televiziune poate distribui publicului mai multe reclame;

**CAMPANII\_MARKETING - ONLINE:** one to one and only one, fiecare campanie de marketing poate avea cel mult o acțiune de promovare online

**CAMPANII\_MARKETING – OFFLINE:** one to one and only one, fiecare campanie de marketing poate avea cel mult o acțiune de promovare offline;

**OFFLINE – LOCALITATI\_TINA:** one to many, fiecare campanie offline de promovare va avea mai multe localități în care se va desfășura;

**LOCALITATI – LOCALITATI\_TINTA:** one to many, o localitate poate face parte din mai multe grupuri de promovare;

**CENTRE\_LOGISTICE - LOCALITATI:** one to one, fiecare centru logistic este situat într-o localitate;

**CENTRE\_LOGISTICE - SPATII\_COMERCIALE:** one and only one to many, un centru logistic va deservi mai multe magazine, însă fiecare magazin se va aproviziona numai de la un singur centru logistic;

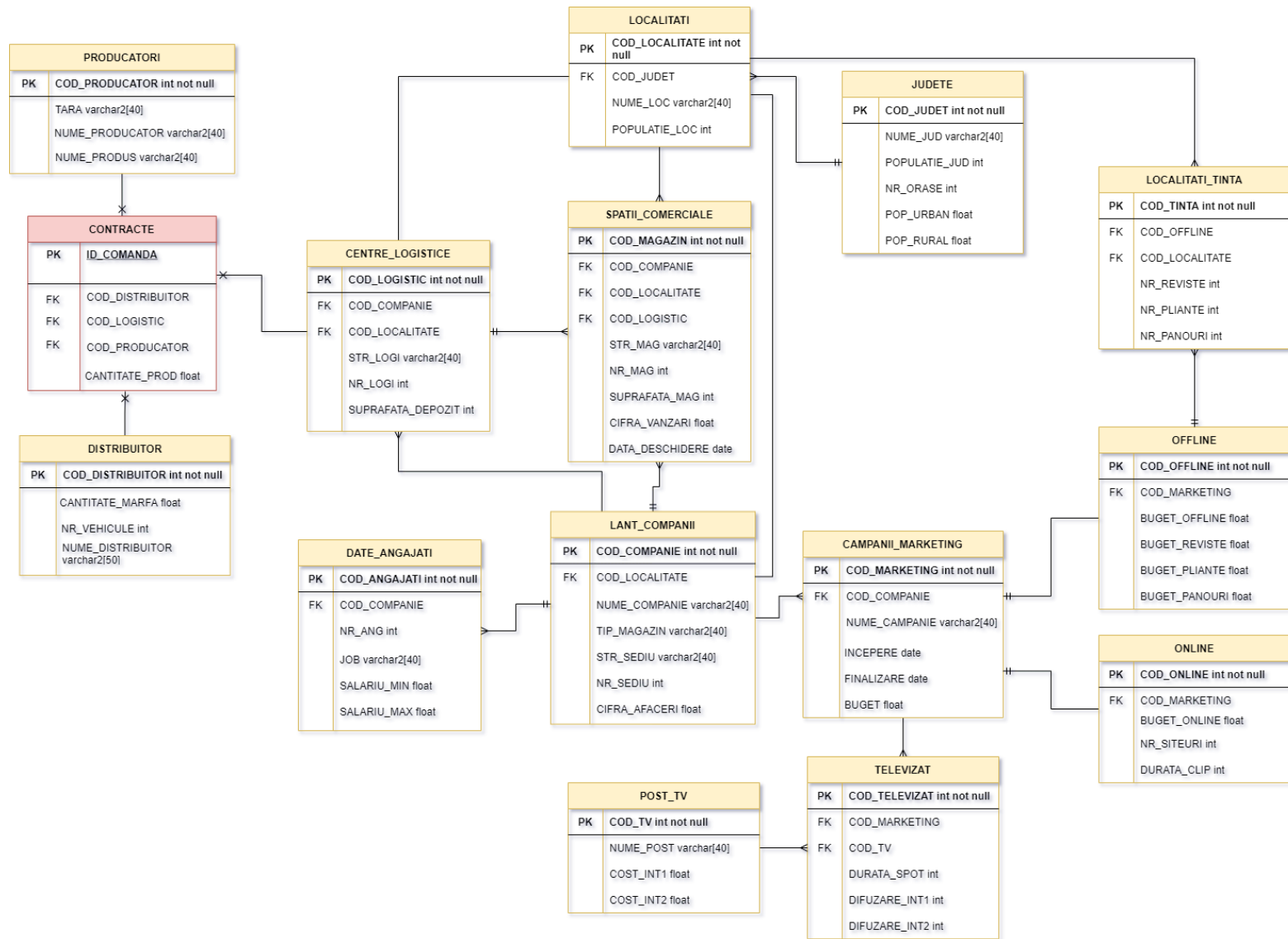
**CENTRE\_LOGISTICE - PRODUCATORI:** many to many, mai multe centre logistice se vor aproviziona de la mai mulți producători;

**CENTRE\_LOGISTICE – LOCALITATI:** one to one, fiecare centru se găsește într-o localitate

**DISTRIBUITOR - CENTRE\_LOGISTICE:** many to many, mai multe centre logistice vor primi marfa de la mai mulți distribuitori;

**DISTRIBUITOR - PRODUCATORI:** many to many, mai mulți producători trimit marfa cu mai mulți distribuitori.

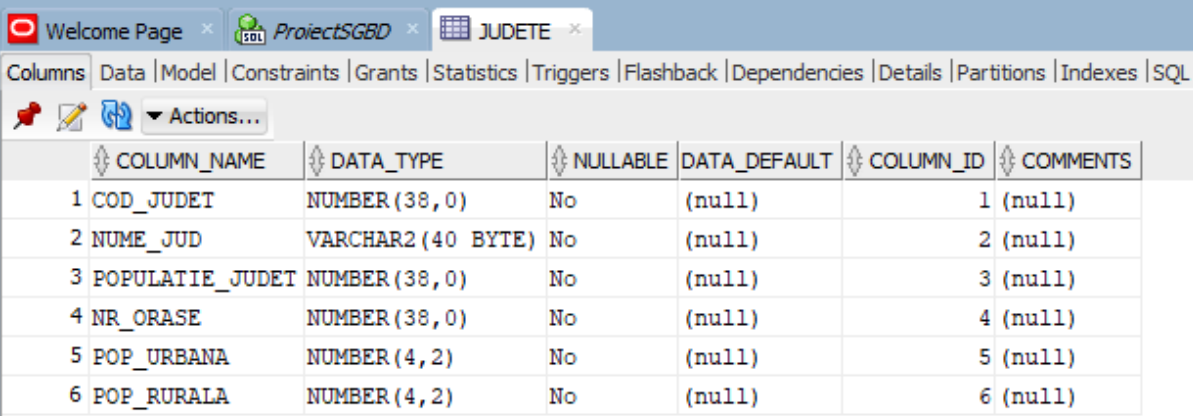
### 3. Diagrama conceptuală



## 4. Implementarea bazei de date

❖ Tabela **JUDEȚE**:

```
CREATE TABLE JUDETE(  
    COD_JUDET int not NULL PRIMARY KEY,  
    NUME_JUD varchar2(40) not NULL,  
    POPULATIE_JUDET int not NULL,  
    NR_ORASE int not NULL,  
    POP_URBANA number(4,2) not NULL, -- populatia urbana procentul  
    POP_RURALA number(4,2) not NULL, -- populatia rurala procentual  
    CHECK (POP_URBANA + POP_RURALA = 100.00)  
);
```

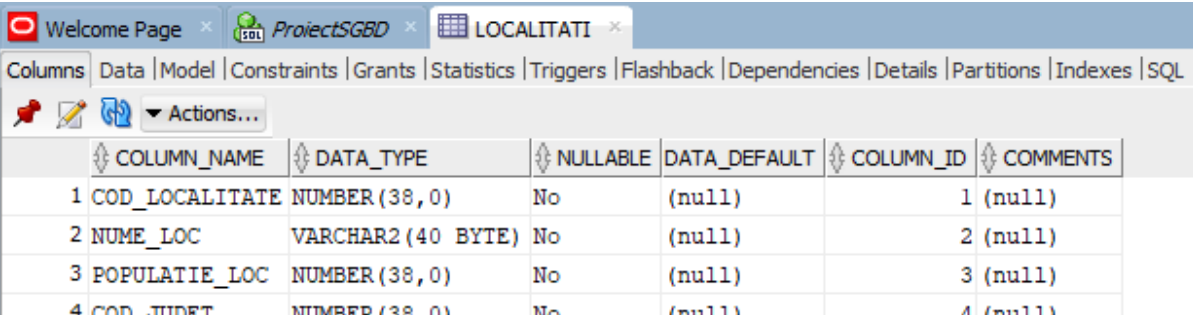


The screenshot shows the SQL Developer interface with the 'JUDETE' table selected. The 'Columns' tab is active, displaying the following table structure:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_JUDET	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_JUD	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	POPULATIE_JUDET	NUMBER(38,0)	No	(null)	3	(null)
4	NR_ORASE	NUMBER(38,0)	No	(null)	4	(null)
5	POP_URBANA	NUMBER(4,2)	No	(null)	5	(null)
6	POP_RURALA	NUMBER(4,2)	No	(null)	6	(null)

❖ Tabela **LOCALITĂȚI**:

```
CREATE TABLE LOCALITATI(  
    COD_LOCALITATE int not NULL PRIMARY KEY,  
    NUME_LOC varchar2(40) not NULL,  
    POPULATIE_LOC int not NULL,  
    COD_JUDET int not NULL,  
    FOREIGN KEY(COD_JUDET) REFERENCES JUDETE(COD_JUDET)  
    -- județul din care face parte localitatea  
);
```

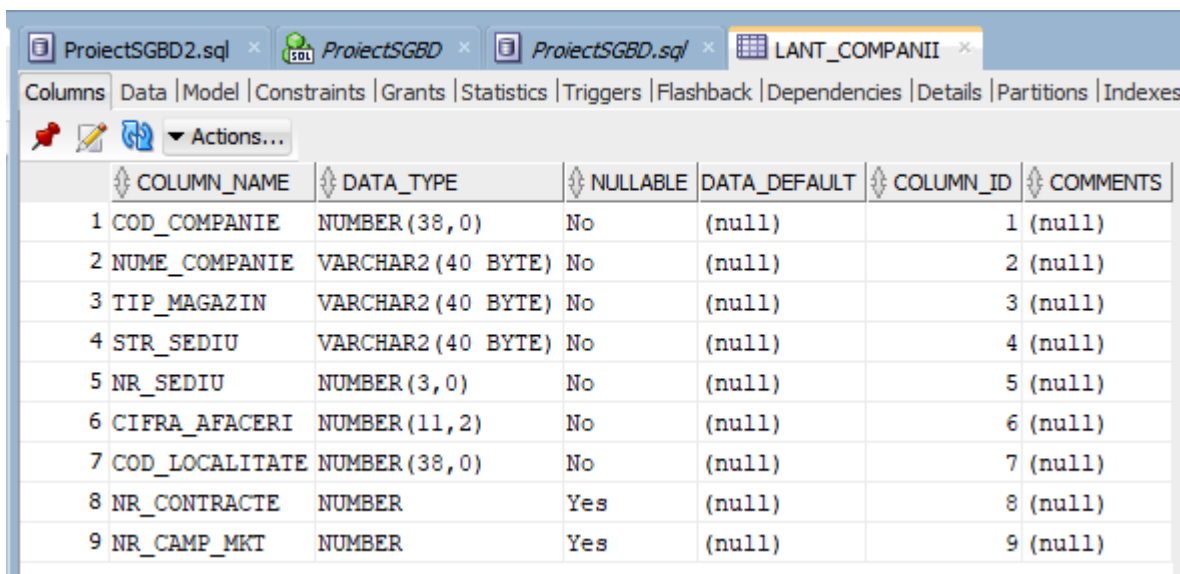


The screenshot shows the SQL Developer interface with the 'LOCALITATI' table selected. The 'Columns' tab is active, displaying the following table structure:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_LOCALITATE	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_LOC	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	POPULATIE_LOC	NUMBER(38,0)	No	(null)	3	(null)
4	COD_JUDET	NUMBER(38,0)	No	(null)	4	(null)

❖ Tabela **LANT\_COMPANII**:

```
CREATE TABLE LANT_COMPANII(
    COD_COMPANIE int not NULL PRIMARY KEY,
    NUME_COMPANIE varchar2(40) not NULL,
    TIP_MAGAZIN varchar2(40) not NULL, -- tipul de magazine deținut de
    companie (bricolaj, discount etc. )
    STR_SEDIU varchar2(40) not NULL, -- strada pe care se află sediul
    central
    NR_SEDIU number (3,0) not NULL, -- numărul la care se află sediul
    central
    CIFRA_AFACERI number(11,2) not NULL,
    COD_LOCALITATE int not NULL,
    NR_CONTRACTE number (5,0),
    NR_CAMP_MKT number (5,0),
    FOREIGN KEY(COD_LOCALITATE) REFERENCES LOCALITATI(COD_LOCALITATE)
    -- localitatea în care se află sediul central
);
```

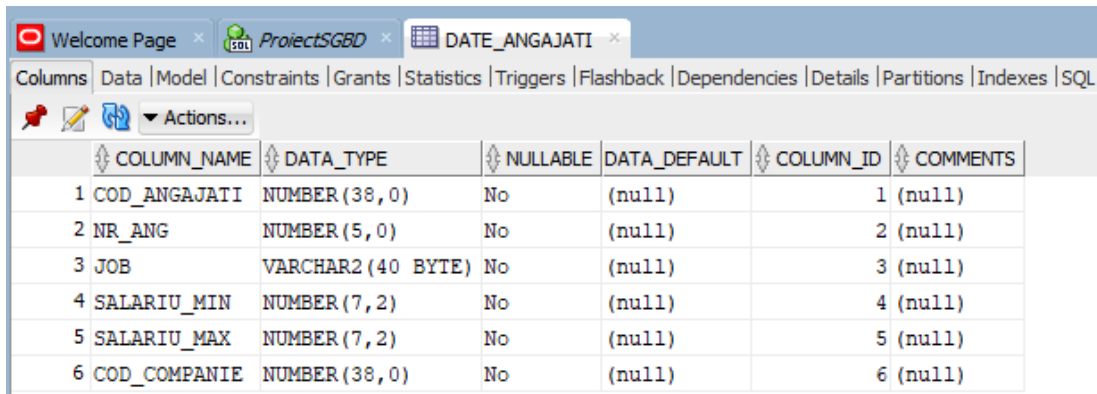


	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_COMPANIE	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_COMPANIE	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	TIP_MAGAZIN	VARCHAR2(40 BYTE)	No	(null)	3	(null)
4	STR_SEDIU	VARCHAR2(40 BYTE)	No	(null)	4	(null)
5	NR_SEDIU	NUMBER(3,0)	No	(null)	5	(null)
6	CIFRA_AFACERI	NUMBER(11,2)	No	(null)	6	(null)
7	COD_LOCALITATE	NUMBER(38,0)	No	(null)	7	(null)
8	NR_CONTRACTE	NUMBER	Yes	(null)	8	(null)
9	NR_CAMP_MKT	NUMBER	Yes	(null)	9	(null)

❖ Tabela **DATE\_ANGAJATI**:

```
CREATE TABLE DATE_ANGAJATI(
    COD_ANGAJATI int not NULL PRIMARY KEY,
    NR_ANG number(5,0) not NULL, -- numărul de angajați
    JOB varchar2(40) not NULL, -- titlul jobului
    SALARIU_MIN number(7,2) not NULL, -- salariul minim pentru job
    SALARIU_MAX number(7,2) not NULL, -- salariul maxim pentru job
    COD_COMPANIE int not NULL,
    FOREIGN KEY(COD_COMPANIE) REFERENCES LANT_COMPANII(COD_COMPANIE)
    -- compania pentru care se rețin datele despre un tip de angajat
);
```

);

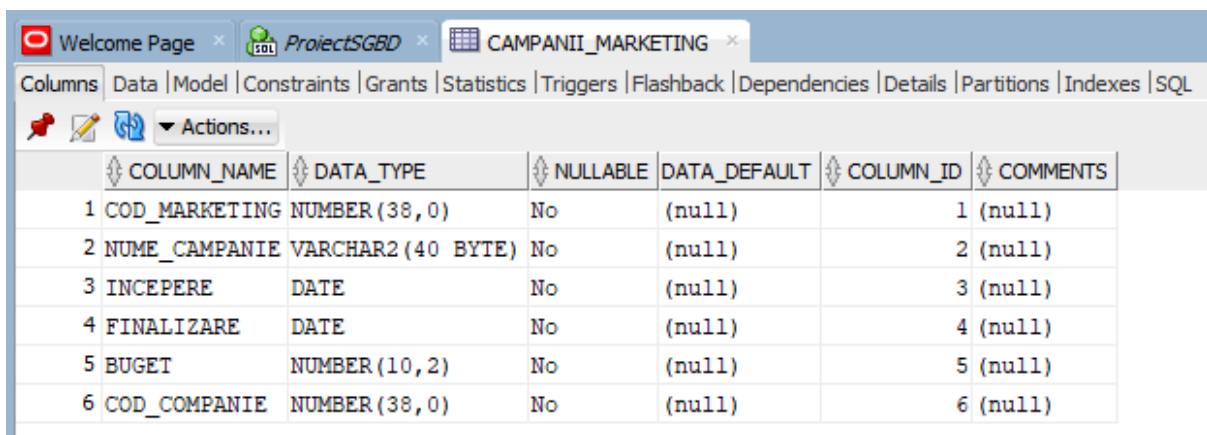


The screenshot shows the SQL Developer interface with the 'DATE\_ANGAJATI' table selected. The table structure is as follows:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_ANGAJATI	NUMBER(38,0)	No	(null)	1	(null)
2	NR_ANG	NUMBER(5,0)	No	(null)	2	(null)
3	JOB	VARCHAR2(40 BYTE)	No	(null)	3	(null)
4	SALARIU_MIN	NUMBER(7,2)	No	(null)	4	(null)
5	SALARIU_MAX	NUMBER(7,2)	No	(null)	5	(null)
6	COD_COMPANIE	NUMBER(38,0)	No	(null)	6	(null)

❖ Tabela **CAMPANII MARKETING**:

```
CREATE TABLE CAMPANII_MARKETING(  
    COD_MARKETING int not NULL PRIMARY KEY,  
    NUME_COMPANIE varchar2(40) not NULL,  
    INCEPERE date not NULL,  
    FINALIZARE date not NULL,  
    BUGET number(10,2) not NULL ,  
    COD_COMPANIE int not NULL,  
    FOREIGN KEY(COD_COMPANIE) REFERENCES LANT_COMPANII(COD_COMPANIE)  
    -- compania pentru care se realizează campania  
);
```



The screenshot shows the SQL Developer interface with the 'CAMPANII\_MARKETING' table selected. The table structure is as follows:

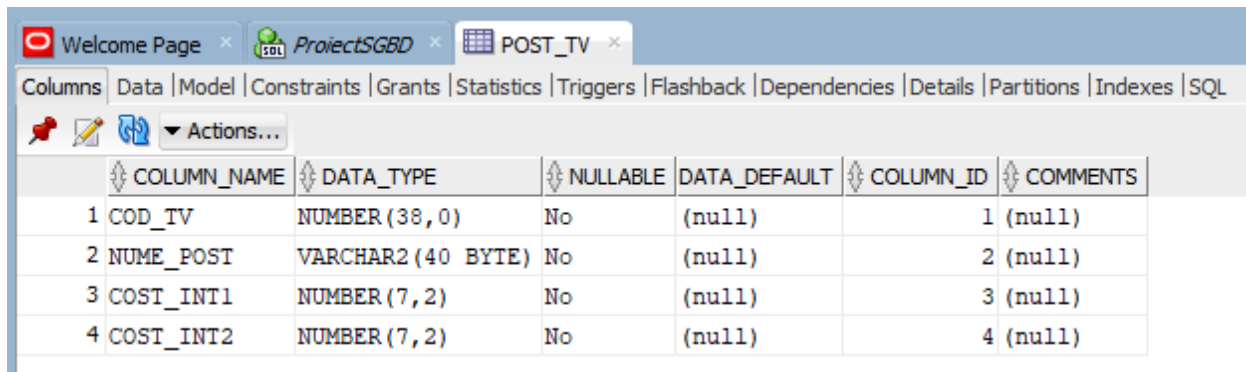
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_MARKETING	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_COMPANIE	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	INCEPERE	DATE	No	(null)	3	(null)
4	FINALIZARE	DATE	No	(null)	4	(null)
5	BUGET	NUMBER(10,2)	No	(null)	5	(null)
6	COD_COMPANIE	NUMBER(38,0)	No	(null)	6	(null)

❖ Tabela **POST TV**:

```
CREATE TABLE POST_TV(  
    COD_TV int not NULL PRIMARY KEY,  
    NUME_POST varchar(40) not NULL,  
    COST_INT1 number(7,2) not NULL, -- cost difuzare în intervalul orar  
    I în € /minut  
    COST_INT2 number(7,2) not NULL -- cost difuzare în intervalul orar  
    II € /minut
```



);

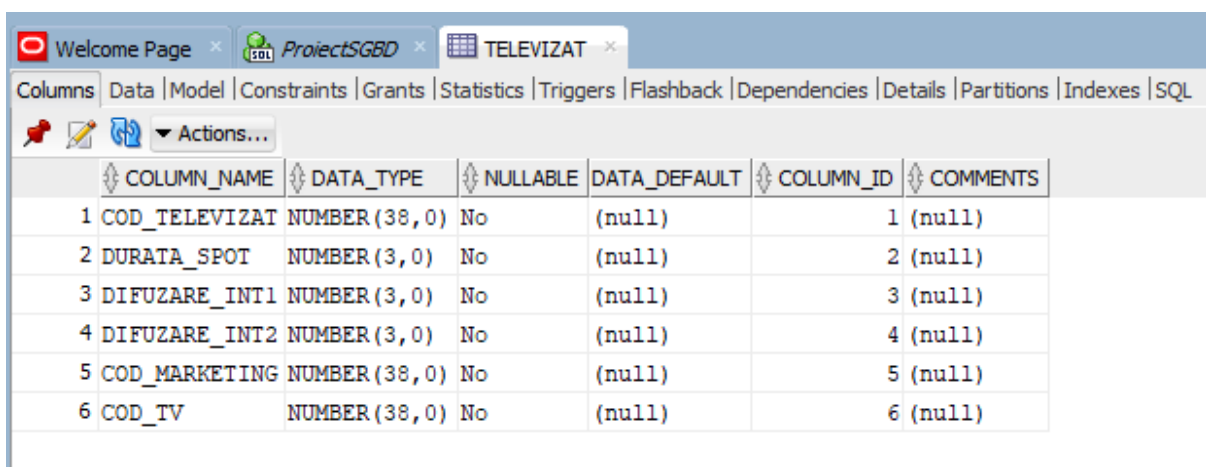


The screenshot shows the SQL Developer interface with the 'POST\_TV' table selected. The 'Columns' tab is active, displaying a table with 7 columns: COLUMN\_NAME, DATA\_TYPE, NULLABLE, DATA\_DEFAULT, COLUMN\_ID, and COMMENTS. The table contains 4 rows of data.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_TV	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_POST	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	COST_INT1	NUMBER(7,2)	No	(null)	3	(null)
4	COST_INT2	NUMBER(7,2)	No	(null)	4	(null)

❖ Tabela **TELEVIZAT**:

```
CREATE TABLE TELEVIZAT(  
    COD_TELEVIZAT int not NULL PRIMARY KEY,  
    DURATA_SPOT number (3,0) not NULL, -- durată reclamă în secunde  
    DIFUZARE_INT1 number (3,0) not NULL, -- timp de difuzare în  
intervalul orar I în minute/săptămână  
    DIFUZARE_INT2 number (3,0) not NULL, -- timp de difuzare în  
intervalul orar II în minute/săptămână  
    COD_MARKETING int not NULL,  
    COD_TV int not NULL,  
    FOREIGN KEY(COD_MARKETING) REFERENCES  
CAMPANII_MARKETING(COD_MARKETING),  
    -- campania de marketing pentru care se realizează acțiunea de  
promovare televizată  
    FOREIGN KEY(COD_TV) REFERENCES POST_TV(COD_TV)  
    -- postul de televiziune contractat  
);
```



The screenshot shows the SQL Developer interface with the 'TELEVIZAT' table selected. The 'Columns' tab is active, displaying a table with 7 columns: COLUMN\_NAME, DATA\_TYPE, NULLABLE, DATA\_DEFAULT, COLUMN\_ID, and COMMENTS. The table contains 6 rows of data.

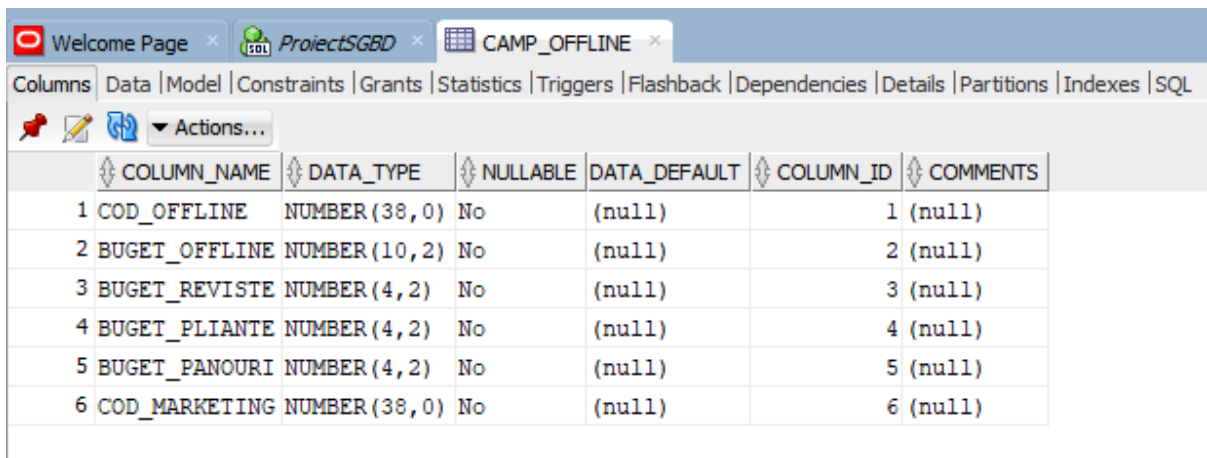
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_TELEVIZAT	NUMBER(38,0)	No	(null)	1	(null)
2	DURATA_SPOT	NUMBER(3,0)	No	(null)	2	(null)
3	DIFUZARE_INT1	NUMBER(3,0)	No	(null)	3	(null)
4	DIFUZARE_INT2	NUMBER(3,0)	No	(null)	4	(null)
5	COD_MARKETING	NUMBER(38,0)	No	(null)	5	(null)
6	COD_TV	NUMBER(38,0)	No	(null)	6	(null)

❖ Tabela **CAMPANII OFFLINE**:

```

CREATE TABLE CAMP_OFFLINE(
    COD_OFFLINE int not NULL PRIMARY KEY,
    BUGET_OFFLINE number(10,2) not NULL,
    BUGET_REVISTE number(4,2) not NULL,
    BUGET_PLIANTE number(4,2) not NULL,
    BUGET_PANOURI number(4,2) not NULL,
    COD_MARKETING int not NULL,
    FOREIGN KEY(COD_MARKETING) REFERENCES
CAMPANII_MARKETING(COD_MARKETING)
    -- campania de marketing pentru care se realizează acțiunea de
    promovare offline
);

```



The screenshot shows a database management interface with a tab for 'CAMP\_OFFLINE'. Below the tab, there is a table with 6 columns: COLUMN\_NAME, DATA\_TYPE, NULLABLE, DATA\_DEFAULT, COLUMN\_ID, and COMMENTS. The rows represent the columns of the table.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_OFFLINE	NUMBER(38,0)	No	(null)	1	(null)
2	BUGET_OFFLINE	NUMBER(10,2)	No	(null)	2	(null)
3	BUGET_REVISTE	NUMBER(4,2)	No	(null)	3	(null)
4	BUGET_PLIANTE	NUMBER(4,2)	No	(null)	4	(null)
5	BUGET_PANOURI	NUMBER(4,2)	No	(null)	5	(null)
6	COD_MARKETING	NUMBER(38,0)	No	(null)	6	(null)

❖ Tabela **CAMPANII ONLINE**:

```

CREATE TABLE CAMP_ONLINE(
    COD_ONLINE int not NULL PRIMARY KEY,
    BUGET_ONLINE number(10,2) not NULL,
    NR_SITEURI number(6,2) not NULL, -- numărul de site-uri partenere
    DURATA_CLIP number(4,2) not NULL,
    COD_MARKETING int not NULL,
    FOREIGN KEY(COD_MARKETING) REFERENCES
CAMPANII_MARKETING(COD_MARKETING)
    -- campania de marketing pentru care se realizează acțiunea de
    promovare online
);

```

Welcome PageProjectSGBDCAMP\_ONLINE

ColumnsDataModelConstraintsGrantsStatisticsTriggersFlashbackDependenciesDetailsPartitionsIndexesSQL

Actions...




	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_ONLINE	NUMBER(38,0)	No	(null)	1	(null)
2	BUGET_ONLINE	NUMBER(10,2)	No	(null)	2	(null)
3	NR_SITEURI	NUMBER(6,2)	No	(null)	3	(null)
4	DURATA_CLIP	NUMBER(4,2)	No	(null)	4	(null)
5	COD_MARKETING	NUMBER(38,0)	No	(null)	5	(null)

❖ Tabela **LOCALITĂȚI ȚINTĂ**:

```
CREATE TABLE LOCALITATI_TINTA (
    COD_TINTA int not NULL PRIMARY KEY,
    NR_REVISTE number(8,0),
    NR_PLIANTE number(8,0),
    NR_PANOURI number(8,0),
    COD_OFFLINE int not NULL,
    COD_LOCALITATE int not NULL,
    FOREIGN KEY(COD_OFFLINE) REFERENCES CAMP_OFFLINE(COD_OFFLINE),
    -- acțiune de promovare offline din care face parte localitate
    FOREIGN KEY(COD_LOCALITATE) REFERENCES LOCALITATI(COD_LOCALITATE)
    -- localitatea parte din campania de promovare
);
```

Welcome Page x ProiectSGBD x LOCALITATI\_TINTA x

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL



Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_TINTA	NUMBER(38,0)	No	(null)	1 (null)	
2	NR_REVISTE	NUMBER(8,0)	Yes	(null)	2 (null)	
3	NR_PLIANTE	NUMBER(8,0)	Yes	(null)	3 (null)	
4	NR_PANOURI	NUMBER(8,0)	Yes	(null)	4 (null)	
5	COD_OFFLINE	NUMBER(38,0)	No	(null)	5 (null)	
6	COD_LOCALITATE	NUMBER(38,0)	No	(null)	6 (null)	

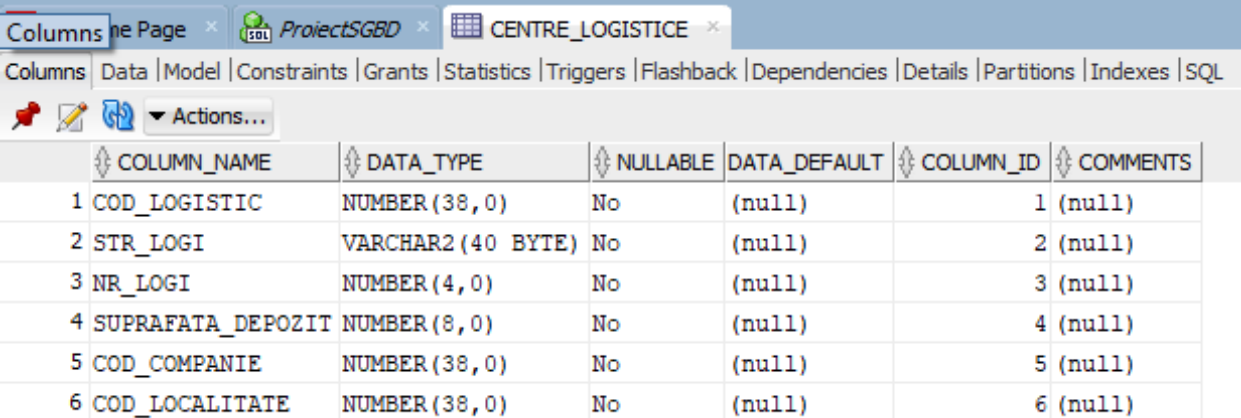
❖ Tabela **CENTRE LOGISTICE**:

```
CREATE TABLE CENTRE_LOGISTICE (
    COD_LOGISTIC int not NULL PRIMARY KEY,
    STR_LOGI varchar2(40) not NULL, -- strada pe care se află depozitul
    NR_LOGI number(4,0) not NULL, -- numărul la care se află depozitul
    SUPRAFATA_DEPOZIT number(8,0) not NULL,
```

```

COD_COMPANIE int not NULL,
COD_LOCALITATE int not NULL,
FOREIGN KEY(COD_LOCALITATE) REFERENCES LOCALITATI(COD_LOCALITATE),
-- localitatea în care se află depozitul
FOREIGN KEY(COD_COMPANIE) REFERENCES LANT_COMPANII(COD_COMPANIE)
-- compania de care aparține depozitul
);

```



The screenshot shows the Oracle SQL Developer interface with the 'Columns' tab selected for the 'CENTRE\_LOGISTICE' table. The table has 6 columns: COD\_LOGISTIC, STR\_LOGI, NR\_LOGI, SUPRAFATA\_DEPOZIT, COD\_COMPANIE, and COD\_LOCALITATE. All columns are of type NUMBER or VARCHAR2, are not nullable, and have a default value of (null).

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_LOGISTIC	NUMBER(38,0)	No	(null)	1	(null)
2	STR_LOGI	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	NR_LOGI	NUMBER(4,0)	No	(null)	3	(null)
4	SUPRAFATA_DEPOZIT	NUMBER(8,0)	No	(null)	4	(null)
5	COD_COMPANIE	NUMBER(38,0)	No	(null)	5	(null)
6	COD_LOCALITATE	NUMBER(38,0)	No	(null)	6	(null)

```

CREATE TABLE SPATII_COMERCIALE(
    COD_MAGAZIN int not NULL PRIMARY KEY,
    STR_MAG varchar2(40)not NULL, -- strada pe care se află magazinul
    NR_MAG number(4,0) not NULL, -- numărul la care se află magazinul
    SUPRAFATA_MAG number(8,0) not NULL,
    CIFRA_VANZARI number(10,2) not NULL,
    DATA_DESCHIDERE date not NULL,
    COD_COMPANIE int not NULL,
    COD_LOCALITATE int not NULL,
    COD_LOGISTIC int not NULL,
    FOREIGN KEY(COD_LOCALITATE) REFERENCES LOCALITATI(COD_LOCALITATE),
    -- localitatea în care se află magazinul
    FOREIGN KEY(COD_COMPANIE) REFERENCES LANT_COMPANII(COD_COMPANIE),
    -- compania de care aparține magazinul
    FOREIGN KEY(COD_LOGISTIC) REFERENCES CENTRE_LOGISTICE(COD_LOGISTIC)
    -- depozitul de la care se aprovizionează magazinul
);

```

Welcome Page

ProjectSGBD

SPATII\_COMERCIALE

Columns

Data

Model

Constraints

Grants

Statistics

Triggers

Flashback




Dependencies

Details

Partitions

Indexes

SQL



Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_MAGAZIN	NUMBER(38,0)	No	(null)	1	(null)
2	STR_MAG	VARCHAR2(40 BYTE)	No	(null)	2	(null)
3	NR_MAG	NUMBER(4,0)	No	(null)	3	(null)
4	SUPRAFATA_MAG	NUMBER(8,0)	No	(null)	4	(null)
5	CIFRA_VANZARI	NUMBER(10,2)	No	(null)	5	(null)
6	DATA_DESCHIDERE	DATE	No	(null)	6	(null)
7	COD_COMPANIE	NUMBER(38,0)	No	(null)	7	(null)
8	COD_LOCALITATE	NUMBER(38,0)	No	(null)	8	(null)
9	COD_LOGISTIC	NUMBER(38,0)	No	(null)	9	(null)

❖ Tabela **PRODUCĂTORI**:

```
CREATE TABLE PRODUCATORI(
    COD_PRODUCATOR int not NULL PRIMARY KEY,
    TARA varchar2(40) not NULL, -- țara unde își desfășoară firma
    activitatea
    NUME_PRODUCATOR varchar2(40) not NULL,
    NUME_PRODUS varchar2(40) not NULL,
    CANTITATE_DISPONIBILA number(10,2) not NULL,
    PRET_UNITAR number(10,2) not NULL
);
```

Welcome Page

ProjectSGBD

PRODUCATORI

Columns

Data

Model

Constraints

Grants

Statistics

Triggers

Flashback




Dependencies

Details

Partitions

Indexes

SQL



Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_PRODUCATOR	NUMBER (38,0)	No	(null)	1 (null)	
2	TARA	VARCHAR2 (40 BYTE)	No	(null)	2 (null)	
3	NUME_PRODUCATOR	VARCHAR2 (40 BYTE)	No	(null)	3 (null)	
4	NUME_PRODUS	VARCHAR2 (40 BYTE)	No	(null)	4 (null)	
5	CANTITATE_DISPONIBILA	NUMBER	No	(null)	5 (null)	
6	PRET_UNITAR	NUMBER	No	(null)	6 (null)	

❖ Tabela **DISTRIBUITORI**:

```
CREATE TABLE DISTRIBUITOR(
```

```

    COD_DISTRIBUTOR int not NULL PRIMARY KEY,
    CANTITATE_MARFA number(8,2) not NULL, -- cantitatea maxima de marfa
    ce poate fi distribuita/comandă
    NR_VEHICULE number(4,0) not NULL, -- numărul de vehicule din flota
    auto
    NUME_DISTRIBUTOR VARCHAR2(50)
);

```

The screenshot shows a database management interface with a tab for 'DISTRIBUTOR'. Below the tab, there is a table with 7 columns: COLUMN\_NAME, DATA\_TYPE, NULLABLE, DATA\_DEFAULT, COLUMN\_ID, and COMMENTS. The table contains 4 rows of data for the DISTRIBUTOR table.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	COD_DISTRIBUTOR	NUMBER(38,0)	No	(null)	1	(null)
2	CANTITATE_MARFA	NUMBER(8,2)	No	(null)	2	(null)
3	NR_VEHICULE	NUMBER(4,0)	No	(null)	3	(null)
4	NUME_DISTRIBUTOR	VARCHAR2(50 BYTE)	Yes	(null)	4	(null)

❖ Tabela **CONTRACTE**:

```

CREATE TABLE CONTRACTE(
    ID_COMANDA int not null PRIMARY KEY,
    CANTITATE_PROD number(8,2) not NULL,
    COD_DISTRIBUTOR int not NULL,
    COD_PRODUCATOR int not NULL,
    COD_LOGISTIC int not NULL,
    INCEPERE date not NULL,
    INCHEIERE date not NULL,
    FOREIGN KEY(COD_DISTRIBUTOR) REFERENCES
DISTRIBUTOR(COD_DISTRIBUTOR),
    FOREIGN KEY(COD_PRODUCATOR) REFERENCES PRODUCATORI(COD_PRODUCATOR),
    FOREIGN KEY(COD_LOGISTIC) REFERENCES CENTRE_LOGISTICE(COD_LOGISTIC)
);

```

Columns						
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_COMANDA	NUMBER(38,0)	No	(null)	1	(null)
2	CANTITATE_PROD	NUMBER(8,2)	No	(null)	2	(null)
3	COD_DISTRIBUTOR	NUMBER(38,0)	No	(null)	3	(null)
4	COD_PRODUCATOR	NUMBER(38,0)	No	(null)	4	(null)
5	COD_LOGISTIC	NUMBER(38,0)	No	(null)	5	(null)
6	INCEPERE	DATE	No	(null)	6	(null)
7	INCHEIERE	DATE	No	(null)	7	(null)

## 5. Popularea tabelor

---

-- Inserare date JUDETE

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (1, 'Iasi', 772348, 4, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (2, 'Prahova', 762886, 5, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (3, 'Cluj', 691106, 7, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (4, 'Constanta', 684082, 6, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (5, 'Timis', 68354, 4, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (6, 'Dolj', 660544, 8, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
```

```
POP_URBANA, POP_RURALA)
VALUES (7, 'Suceava', 63481, 5, 40, 60);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (8, 'Bacău', 616168, 6, 50, 50);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (9, 'Arges', 612431, 7, 60, 40);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (10, 'Bihor', 575398, 8, 40, 60);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (11, 'Mures', 540508, 8, 50, 50);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (12, 'Galati', 530612, 11, 60, 40);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (13, 'Brasov', 529906, 8, 40, 60);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (14, 'Dâmbovita', 510287, 8, 50, 50);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (15, 'Neamt', 507399, 11, 60, 40);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (16, 'Maramures', 472117, 12, 40, 60);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (17, 'Buzău', 440347, 5, 50, 50);

INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (18, 'Olt', 421769, 8, 60, 40);
```



```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (19, 'Arad', 412235, 9, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (20, 'Hunedoara', 410383, 8, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (21, 'Botosani', 404429, 8, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (22, 'Vaslui', 39334, 5, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (23, 'Sibiu', 377273, 7, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (24, 'Vâlcea', 37424, 11, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (25, 'Teleorman', 369897, 8, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (26, 'Ilfov', 353481, 9, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (27, 'Gorj', 345771, 11, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (28, 'Alba', 342336, 11, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,  
POP_URBANA, POP_RURALA)  
VALUES (29, 'Vrancea', 33951, 5, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
```

```
POP_URBANA, POP_RURALA)
VALUES (30, 'Satu Mare', 336117, 5, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (31, 'Brăila', 316652, 6, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (32, 'Harghita', 304765, 3, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (33, 'Bistrita-Năsăud', 301425, 4, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (34, 'Caras-Severin', 287535, 5, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (35, 'Călărași', 287269, 7, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (36, 'Giurgiu', 267147, 6, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (37, 'Ialomita', 265559, 5, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (38, 'Mehedinti', 259212, 3, 50, 50);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (39, 'Sălaj', 225631, 5, 60, 40);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (40, 'Tulcea', 211622, 4, 40, 60);
```

```
INSERT INTO JUDETE (COD_JUDET, NUME_JUD, POPULATIE_JUDET, NR_ORASE,
POP_URBANA, POP_RURALA)
VALUES (41, 'Covasna', 22375, 7, 50, 50);
```

Welcome Page						
ProiectSGBD						
JUDETE						
Columns	Data	Model	Constraints	Grants	Statistics	Triggers
					Flashback	Dependencies
						Details
						Partitions
						Indexes
						SQL
Sort.. Filter:						
	COD_JUDET	NUME_JUD	POPULATIE_JUDET	NR_ORASE	POP_URBANA	POP_RURALA
1	1	Iasi	772348	4	40	60
2	2	Prahova	762886	5	50	50
3	3	Cluj	691106	7	60	40
4	4	Constanta	684082	6	40	60
5	5	Timis	68354	4	50	50
6	6	Dolj	660544	8	60	40
7	7	Suceava	63481	5	40	60
8	8	Bacău	616168	6	50	50
9	9	Arges	612431	7	60	40
10	10	Bihor	575398	8	40	60
11	11	Mures	540508	8	50	50
12	12	Galati	530612	11	60	40
13	13	Brasov	529906	8	40	60
14	14	Dâmbovita	510287	8	50	50
15	15	Neamt	507399	11	60	40
16	16	Maramures	472117	12	40	60
17	17	Buzău	440347	5	50	50
18	18	Olt	421769	8	60	40
19	19	Arad	412235	9	40	60
20	20	Hunedoara	410383	8	50	50
21	21	Botosani	404429	8	60	40
22	22	Vaslui	39334	5	40	60
23	23	Sibiu	377273	7	50	50
24	24	Vâlcea	37424	11	60	40
25	25	Teleorman	369897	8	40	60
26	26	Ilfov	353481	9	50	50
27	27	Gorj	345771	11	60	40
28	28	Alba	342336	11	40	60
29	29	Vrancea	33951	5	50	50
30	30	Satu Mare	336117	5	60	40
31	31	Brăila	316652	6	40	60
32	32	Harghita	304765	3	50	50
33	33	Bistrita-Năsăud	301425	4	60	40
34	34	Caras-Severin	287535	5	40	60
35	35	Călărași	287269	7	50	50
36	36	Giurgiu	267147	6	60	40
37	37	Ialomita	265559	5	40	60
38	38	Mehedinti	259212	3	50	50
39	39	Sălaj	225631	5	60	40
40	40	Tulcea	211622	4	40	60

-- Inserare date LOCALITATI

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (1, 'Targoviste', 90000, 14);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (2, 'Ploiesti', 200000, 2);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (3, 'Moreni', 20000, 14);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (4, 'Pucioasa', 14000, 14);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (5, 'Timisoara', 300000, 4);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

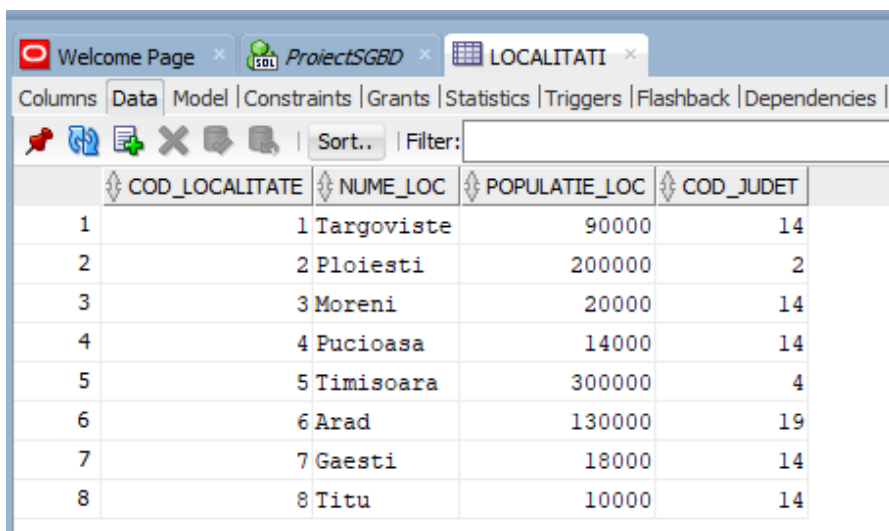
```
VALUES (6, 'Arad', 130000, 19);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (7, 'Gaesti', 18000, 14);
```

```
INSERT INTO LOCALITATI (COD_LOCALITATE, NUME_LOC, POPULATIE_LOC,  
COD_JUDET)
```

```
VALUES (8, 'Titu', 10000, 14);
```



The screenshot shows a database management interface with a table named 'LOCALITATI'. The table has four columns: 'COD\_LOCALITATE', 'NUME\_LOC', 'POPULATIE\_LOC', and 'COD\_JUDET'. The data is as follows:

	COD_LOCALITATE	NUME_LOC	POPULATIE_LOC	COD_JUDET
1	1	Targoviste	90000	14
2	2	Ploiesti	200000	2
3	3	Moreni	20000	14
4	4	Pucioasa	14000	14
5	5	Timisoara	300000	4
6	6	Arad	130000	19
7	7	Gaesti	18000	14
8	8	Titu	10000	14

```
-- Inserare date LANT COMPANII
```

```
INSERT INTO LANT_COMPANII (COD_COMPANIE, NUME_COMPANIE, TIP_MAGAZIN,  
STR_SEDIU, NR_SEDIU, CIFRA_AFACERI, COD_LOCALITATE, NR_CONTRACTE,  
NR_CAMP_MKT)  
VALUES (1, 'Kaufland', 'discount', 'Trandafirilor', 23, 1200000, 5, 4,  
2);
```

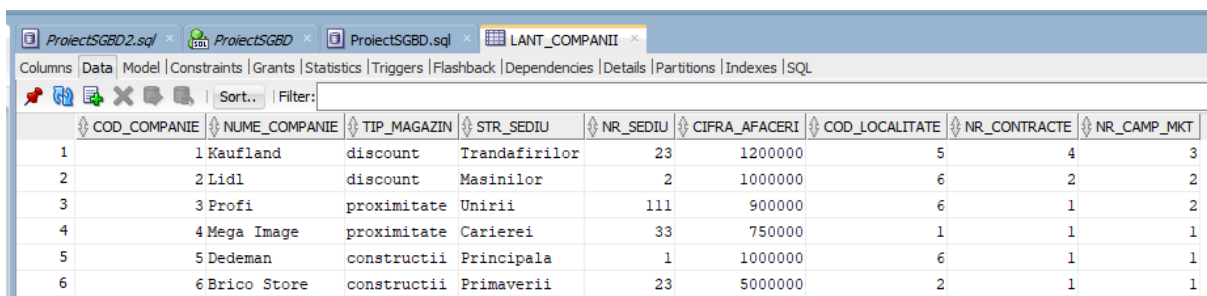
```
INSERT INTO LANT_COMPANII (COD_COMPANIE, NUME_COMPANIE, TIP_MAGAZIN,  
STR_SEDIU, NR_SEDIU, CIFRA_AFACERI, COD_LOCALITATE , NR_CONTRACTE,  
NR_CAMP_MKT)  
VALUES (2, 'Lidl', 'discount', 'Masinilor', 2, 1000000, 6, 3, 1, 2);
```

```
INSERT INTO LANT_COMPANII (COD_COMPANIE, NUME_COMPANIE, TIP_MAGAZIN,  
STR_SEDIU, NR_SEDIU, CIFRA_AFACERI, COD_LOCALITATE , NR_CONTRACTE,  
NR_CAMP_MKT)  
VALUES (3, 'Profi', 'proximitate', 'Unirii', 111, 900000, 6, 1, 1, 2);
```

```
INSERT INTO LANT_COMPANII (COD_COMPANIE, NUME_COMPANIE, TIP_MAGAZIN,  
STR_SEDIU, NR_SEDIU, CIFRA_AFACERI, COD_LOCALITATE , NR_CONTRACTE,  
NR_CAMP_MKT)  
VALUES (4, 'Mega Image', 'proximitate', 'Carieriei', 33, 750000, 1, 1,  
1);
```

```
INSERT INTO LANT_COMPANII (COD_COMPANIE, NUME_COMPANIE, TIP_MAGAZIN,  
STR_SEDIU, NR_SEDIU, CIFRA_AFACERI, COD_LOCALITATE , NR_CONTRACTE,  
NR_CAMP_MKT)  
VALUES (5, 'Dedeman', 'constructii', 'Principala', 1, 1000000, 6, 1, 1);
```

```
INSERT INTO LANT_COMPANII (COD_COMPANIE, NUME_COMPANIE, TIP_MAGAZIN,  
STR_SEDIU, NR_SEDIU, CIFRA_AFACERI, COD_LOCALITATE , NR_CONTRACTE,  
NR_CAMP_MKT)  
VALUES (6, 'Brico Store', 'constructii', 'Primaveriei', 23, 5000000, 2,  
1, 1);
```



The screenshot shows a database management interface with a table named 'LANT\_COMPANII'. The table has 10 columns: COD\_COMPANIE, NUME\_COMPANIE, TIP\_MAGAZIN, STR\_SEDIU, NR\_SEDIU, CIFRA\_AFACERI, COD\_LOCALITATE, NR\_CONTRACTE, and NR\_CAMP\_MKT. There are 6 rows of data displayed.

	COD_COMPANIE	NUME_COMPANIE	TIP_MAGAZIN	STR_SEDIU	NR_SEDIU	CIFRA_AFACERI	COD_LOCALITATE	NR_CONTRACTE	NR_CAMP_MKT
1	1	Kaufland	discount	Trandafirilor	23	1200000	5	4	3
2	2	Lidl	discount	Masinilor	2	1000000	6	2	2
3	3	Profi	proximitate	Unirii	111	900000	6	1	2
4	4	Mega Image	proximitate	Carieriei	33	750000	1	1	1
5	5	Dedeman	constructii	Principala	1	1000000	6	1	1
6	6	Brico Store	constructii	Primaveriei	23	5000000	2	1	1

-- Inserare date CAMPANII MARKETING

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (1, 'Lunea la Lidl', to_date('10.05.2020', 'DD/MM/YYYY'),  
to_date('10.05.2021', 'DD/MM/YYYY'), 1000000, 2);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (2, 'Kauflandtastic', to_date('19.10.2020', 'DD/MM/YYYY'),  
to_date('19.10.2021', 'DD/MM/YYYY'), 123456, 1);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (3, 'Mega Aproape', to_date('10.10.2019', 'DD/MM/YYYY'),  
to_date('10.10.2020', 'DD/MM/YYYY'), 2790000, 4);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (4, 'Briconstructii', to_date('05.07.2018', 'DD/MM/YYYY'),  
to_date('10.10.2019', 'DD/MM/YYYY'), 1234533, 6);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (5, 'Kaufland e ieftin', to_date('03.05.2020', 'DD/MM/YYYY'),  
to_date('03.05.2021', 'DD/MM/YYYY'), 3456781, 1);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (6, 'Hai la Profi', to_date('20.11.2019', 'DD/MM/YYYY'),  
to_date('20.11.2021', 'DD/MM/YYYY'), 230523, 3);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (7, 'Dedeman, mereu aproape', to_date('13.12.2020',  
'DD/MM/YYYY'), to_date('13.07.2021', 'DD/MM/YYYY'), 1300997, 5);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (8, 'Profi langa tine', to_date('16.12.2020', 'DD/MM/YYYY'),  
to_date('11.03.2022', 'DD/MM/YYYY'), 1110000, 3);
```

```
INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE, FINALIZARE, BUGET, COD_COMPANIE)
```

```
VALUES (9, 'Kaufland TV', to_date('11.11.2020', 'DD/MM/YYYY'),  
to_date('11.11.2022', 'DD/MM/YYYY'), 2324449, 1);
```

```

INSERT INTO CAMPANII_MARKETING (COD_MARKETING, NUME_CAMPANIE, INCEPERE,
FINALIZARE, BUGET, COD_COMPANIE)
VALUES (10, 'Lidl TV', to_date('19.04.2019', 'DD/MM/YYYY'),
to_date('19.04.2020', 'DD/MM/YYYY'), 2222222, 2);

```

CAMPANII_MARKETING						
	COD_MARKETING	NUME_CAMPANIE	INCEPERE	FINALIZARE	BUGET	COD_COMPANIE
1	1	Lunea la Lidl	10-MAY-20	10-MAY-21	1000000	2
2	2	Kauflandtastic	19-OCT-20	19-OCT-21	123456	1
3	3	Mega Aproape	10-OCT-19	10-OCT-20	2790000	4
4	4	Bricconstructii	05-JUL-18	10-OCT-19	1234533	6
5	5	Kaufland e ieftin	03-MAY-20	03-MAY-21	3456781	1
6	6	Hai la Profi	20-NOV-19	20-NOV-21	230523	3
7	7	Dedeman, mereu aproape	13-DEC-20	13-JUL-21	1300997	5
8	8	Profi langa tine	16-DEC-20	11-MAR-22	1110000	3
9	9	Kaufland TV	11-NOV-20	11-NOV-22	2324449	1
10	10	Lidl TV	19-APR-19	19-APR-20	2222222	2

-- Inserare date POSTURI TELEVIZIUNE

```

INSERT INTO POST_TV (COD_TV, NUME_POST, COST_INT1, COST_INT2)
VALUES (1, 'PRO TV', 500, 750);

```

```

INSERT INTO POST_TV (COD_TV, NUME_POST, COST_INT1, COST_INT2)
VALUES (2, 'DIGI 24', 350, 600);

```

```

INSERT INTO POST_TV (COD_TV, NUME_POST, COST_INT1, COST_INT2)
VALUES (3, 'NATIONAL GEOGRAPHIC', 150, 300);

```

```

INSERT INTO POST_TV (COD_TV, NUME_POST, COST_INT1, COST_INT2)
VALUES (4, 'DISCOVERY', 200, 290);

```

```

INSERT INTO POST_TV (COD_TV, NUME_POST, COST_INT1, COST_INT2)
VALUES (5, 'TVR1', 350, 550);

```

Welcome Page × ProjectSGBD × POST_TV ×				
Columns Data Model Constraints Grants Statistics Triggers Flashback Deper				
Sort.. Filter:				
	COD_TV	NUME_POST	COST_INT1	COST_INT2
1	1	PRO TV	500	750
2	2	DIGI 24	350	600
3	3	NATIONAL GEOGRPHIC	150	300
4	4	DISCOVERY	200	290
5	5	TVR1	350	550

-- Inserare date CAMPANII TELEVIZIUNE

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (1, 30, 23, 50, 9, 1);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (2, 30, 19, 77, 9, 2);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (3, 30, 12, 34, 9, 5);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (4, 45, 33, 45, 10, 3);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (5, 45, 7, 33, 10, 4);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (6, 29, 23, 55, 7, 1);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (7, 29, 33, 56, 7, 3);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (8, 29, 40, 50, 7, 5);
```



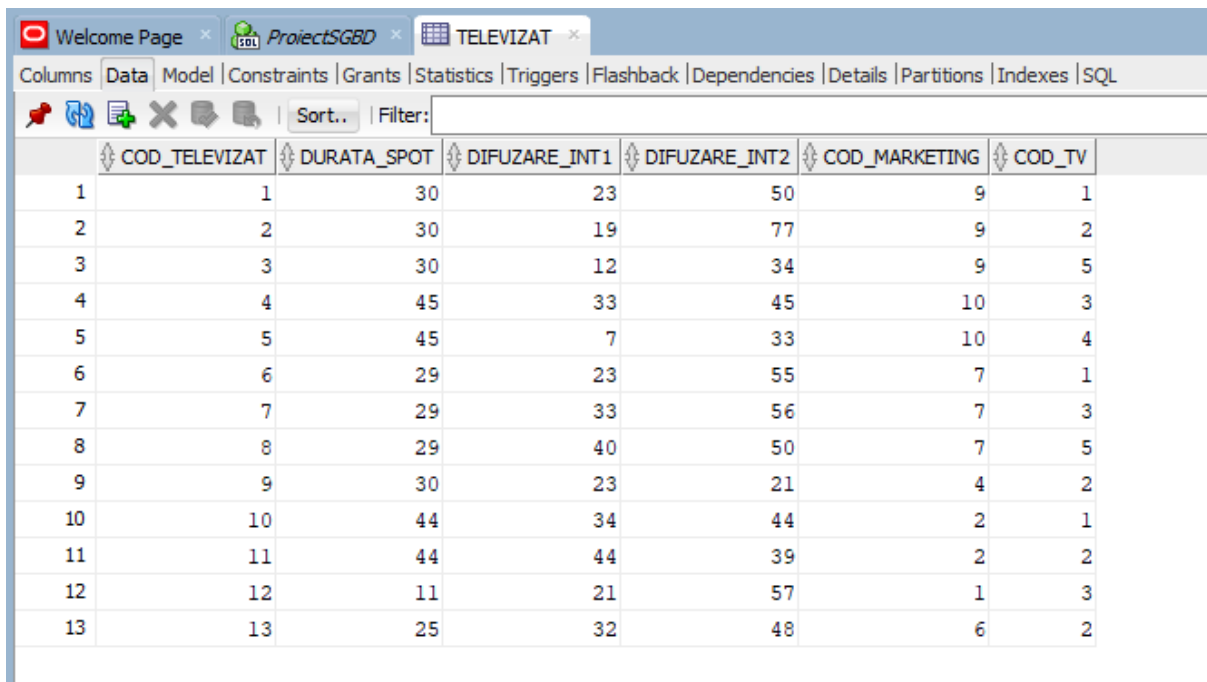
```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (9, 30, 23, 21, 4, 2);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (10, 44, 34, 44, 2, 1);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (11, 44, 44, 39, 2, 2);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (12, 11, 21, 57, 1, 3);
```

```
INSERT INTO TELEVIZAT (COD_TELEVIZAT, DURATA_SPOT, DIFUZARE_INT1,
DIFUZARE_INT2, COD_MARKETING, COD_TV)
VALUES (13, 25, 32, 48, 6, 2);
```



The screenshot shows a database management interface with a tab labeled 'TELEVIZAT'. Below the tab is a table with 7 columns: COD\_TELEVIZAT, DURATA\_SPOT, DIFUZARE\_INT1, DIFUZARE\_INT2, COD\_MARKETING, and COD\_TV. The table contains 13 rows of data, numbered 1 to 13 in the first column. The data values are as follows:

	COD_TELEVIZAT	DURATA_SPOT	DIFUZARE_INT1	DIFUZARE_INT2	COD_MARKETING	COD_TV
1	1	30	23	50	9	1
2	2	30	19	77	9	2
3	3	30	12	34	9	5
4	4	45	33	45	10	3
5	5	45	7	33	10	4
6	6	29	23	55	7	1
7	7	29	33	56	7	3
8	8	29	40	50	7	5
9	9	30	23	21	4	2
10	10	44	34	44	2	1
11	11	44	44	39	2	2
12	12	11	21	57	1	3
13	13	25	32	48	6	2

-- Inserare date CAMPANII ONLINE

```
INSERT INTO CAMP_ONLINE (COD_ONLINE, BUGET_ONLINE, NR_SITEURI,
DURATA_CLIP, COD_MARKETING)
VALUES (1, 1234124, 23, 23, 3);
```

```
INSERT INTO CAMP_ONLINE (COD_ONLINE, BUGET_ONLINE, NR_SITEURI,
```

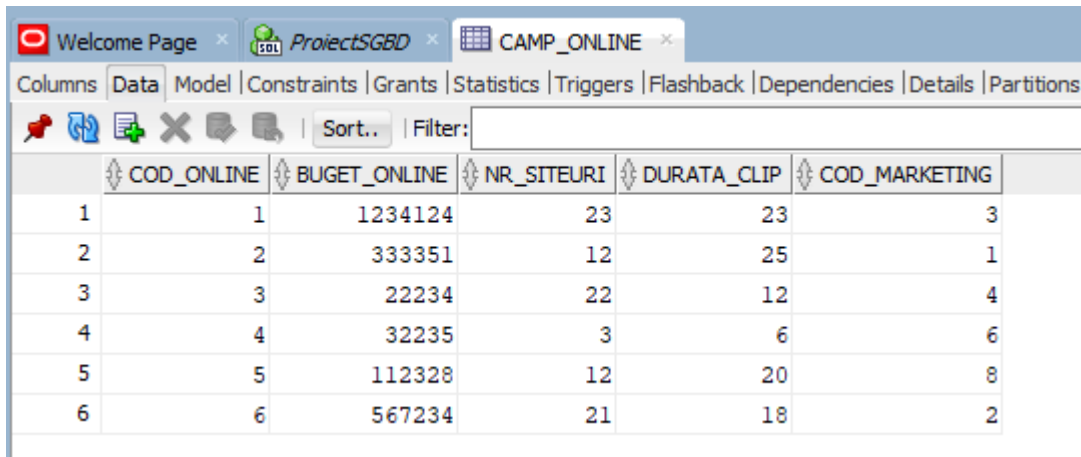
```
DURATA_CLIP, COD_MARKETING)
VALUES (2, 333351, 12, 25, 1);
```

```
INSERT INTO CAMP_ONLINE (COD_ONLINE, BUGET_ONLINE, NR_SITEURI,
DURATA_CLIP, COD_MARKETING)
VALUES (3, 22234, 22, 12, 4);
```

```
INSERT INTO CAMP_ONLINE (COD_ONLINE, BUGET_ONLINE, NR_SITEURI,
DURATA_CLIP, COD_MARKETING)
VALUES (4, 32235, 3, 6, 6);
```

```
INSERT INTO CAMP_ONLINE (COD_ONLINE, BUGET_ONLINE, NR_SITEURI,
DURATA_CLIP, COD_MARKETING)
VALUES (5, 112328, 12, 20, 8);
```

```
INSERT INTO CAMP_ONLINE (COD_ONLINE, BUGET_ONLINE, NR_SITEURI,
DURATA_CLIP, COD_MARKETING)
VALUES (6, 567234, 21, 18, 2);
```



	COD_ONLINE	BUGET_ONLINE	NR_SITEURI	DURATA_CLIP	COD_MARKETING
1	1	1234124	23	23	3
2	2	333351	12	25	1
3	3	22234	22	12	4
4	4	32235	3	6	6
5	5	112328	12	20	8
6	6	567234	21	18	2

```
-- Inserare date CAMPANII OFFLINE
```

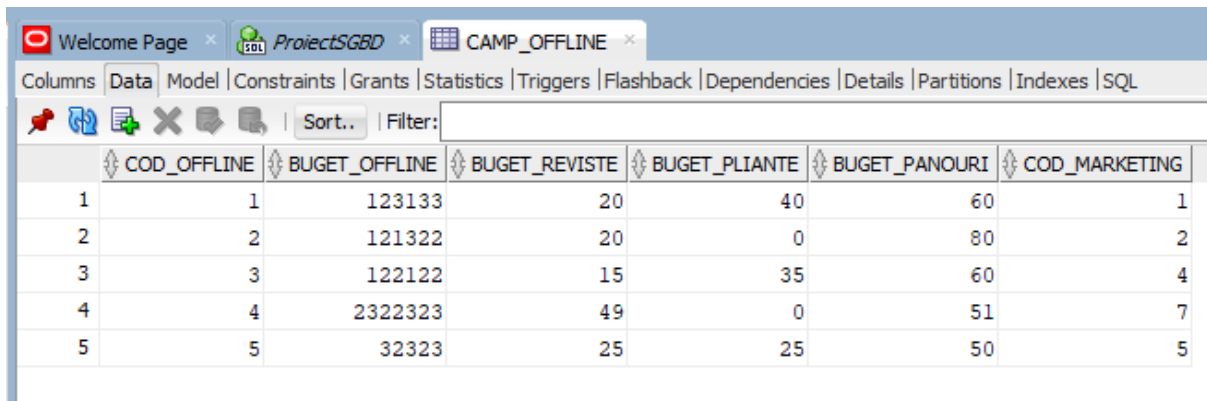
```
INSERT INTO CAMP_OFFLINE (COD_OFFLINE, BUGET_OFFLINE, BUGET_REVISTE,
BUGET_PLIANTE, BUGET_PANOURI, COD_MARKETING)
VALUES (1, 123133, 20, 40, 60, 1);
```

```
INSERT INTO CAMP_OFFLINE (COD_OFFLINE, BUGET_OFFLINE, BUGET_REVISTE,
BUGET_PLIANTE, BUGET_PANOURI, COD_MARKETING)
VALUES (2, 121322, 20, 0, 80, 2);
```

```
INSERT INTO CAMP_OFFLINE (COD_OFFLINE, BUGET_OFFLINE, BUGET_REVISTE,
BUGET_PLIANTE, BUGET_PANOURI, COD_MARKETING)
VALUES (3, 122122, 15, 35, 60, 4);
```

```
INSERT INTO CAMP_OFFLINE (COD_OFFLINE, BUGET_OFFLINE, BUGET_REVISTE,
BUGET_PLIANTE, BUGET_PANOURI, COD_MARKETING)
VALUES (4, 2322323, 49, 0, 51, 7);
```

```
INSERT INTO CAMP_OFFLINE (COD_OFFLINE, BUGET_OFFLINE, BUGET_REVISTE,
BUGET_PLIANTE, BUGET_PANOURI, COD_MARKETING)
VALUES (5, 32323, 25, 25, 50, 5);
```



The screenshot shows a database management interface with a tab for 'CAMP\_OFFLINE'. The table has 6 columns: COD\_OFFLINE, BUGET\_OFFLINE, BUGET\_REVISTE, BUGET\_PLIANTE, BUGET\_PANOURI, and COD\_MARKETING. There are 5 rows of data displayed.

	COD_OFFLINE	BUGET_OFFLINE	BUGET_REVISTE	BUGET_PLIANTE	BUGET_PANOURI	COD_MARKETING
1	1	123133	20	40	60	1
2	2	121322	20	0	80	2
3	3	122122	15	35	60	4
4	4	2322323	49	0	51	7
5	5	32323	25	25	50	5

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (1, 22233, NULL, 12, 2, 5);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (2, 3000, NULL, 23, 2, 3);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (3, 5000, NULL, 12, 4, 4);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (4, 4500, NULL, 12, 4, 3);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (5, 67000, NULL, 22, 4, 6);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (6, 8500, NULL, 11, 4, 2);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)
VALUES (7, 6900, NULL, 22, 4, 1);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (8, 11000, 2323, 12, 1, 2);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (9, 19000, 2222, 12, 1, 3);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (10, 2100, 33333, 3, 3, 4);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (11, 21212, 2222, 21, 3, 6);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (12, 455553, 323, 2, 2, 5);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (13, 53333, 12121, 1, 5, 1);
```

```
INSERT INTO LOCALITATI_TINTA (COD_TINTA, NR_REVISTE, NR_PLIANTE,  
NR_PANOURI, COD_OFFLINE, COD_LOCALITATE)  
VALUES (14, 44423, 222, 23, 5, 5);
```

LOCALITATI_TINTA						
	COD_TINTA	NR_REVISTE	NR_PLIANTE	NR_PANOURI	COD_OFFLINE	COD_LOCALITATE
1	1	22233	(null)	12	2	5
2	2	3000	(null)	23	2	3
3	3	5000	(null)	12	4	4
4	4	4500	(null)	12	4	3
5	5	67000	(null)	22	4	6
6	6	8500	(null)	11	4	2
7	7	6900	(null)	22	4	1
8	8	11000	2323	12	1	2
9	9	19000	2222	12	1	3
10	10	2100	33333	3	3	4
11	11	21212	2222	21	3	6
12	12	455553	323	2	2	5
13	13	53333	12121	1	5	1
14	14	44423	222	23	5	5

-- Inserare date ANGAJATI

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (1, 25000, 'vanzator', 1900, 2839, 1);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (2, 30000, 'vanzator', 1850, 2845, 2);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (3, 23000, 'vanzator', 1650, 2935, 3);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (4, 8000, 'vanzator', 1700, 2771, 4);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (5, 7500, 'vanzator', 1750, 2577, 5);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (6, 9000, 'vanzator', 1790, 2682, 6);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (7, 1000, 'manager', 3141, 4074, 1);
```

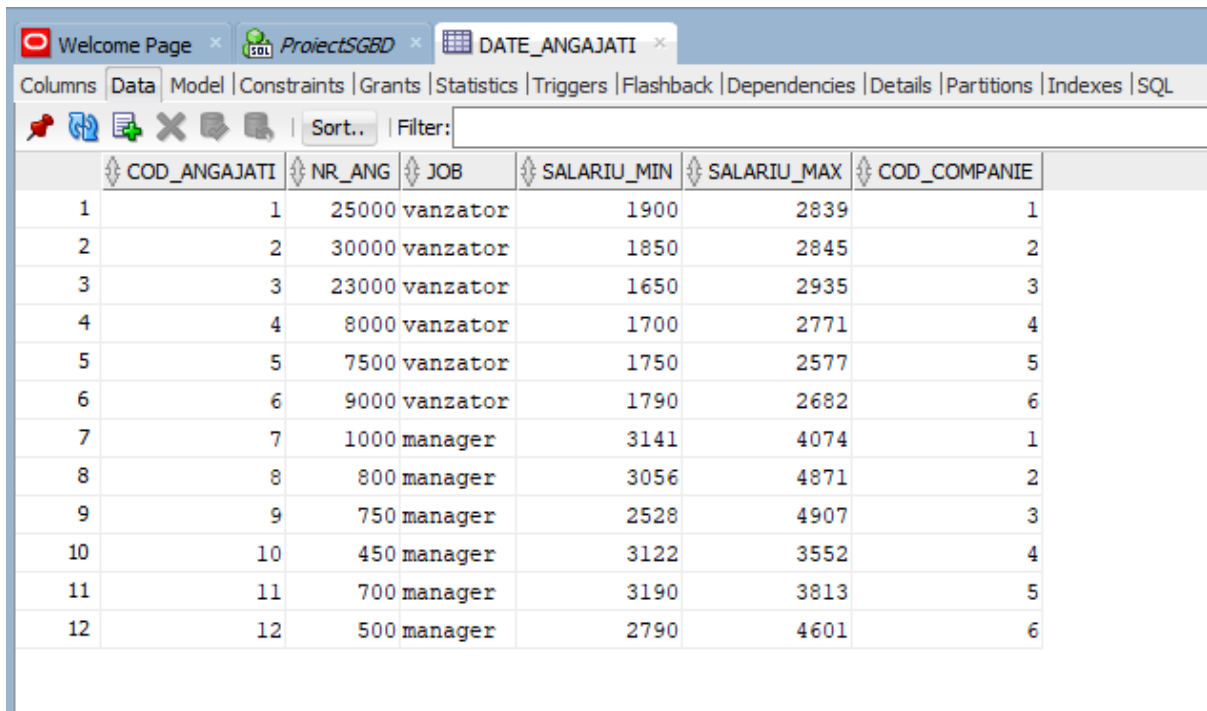
```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (8, 800, 'manager', 3056, 4871, 2);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (9, 750, 'manager', 2528, 4907, 3);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (10, 450, 'manager', 3122, 3552, 4);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (11, 700, 'manager', 3190, 3813, 5);
```

```
INSERT INTO DATE_ANGAJATI (COD_ANGAJATI, NR_ANG, JOB, SALARIU_MIN,
SALARIU_MAX, COD_COMPANIE)
VALUES (12, 500, 'manager', 2790, 4601, 6);
```



The screenshot shows a database management interface with a tab for 'DATE\_ANGAJATI'. The 'Data' tab is selected, displaying a table with 12 rows. The columns are COD\_ANGAJATI, NR\_ANG, JOB, SALARIU\_MIN, SALARIU\_MAX, and COD\_COMPANIE. The data is as follows:

	COD_ANGAJATI	NR_ANG	JOB	SALARIU_MIN	SALARIU_MAX	COD_COMPANIE
1	1	25000	vanzator	1900	2839	1
2	2	30000	vanzator	1850	2845	2
3	3	23000	vanzator	1650	2935	3
4	4	8000	vanzator	1700	2771	4
5	5	7500	vanzator	1750	2577	5
6	6	9000	vanzator	1790	2682	6
7	7	1000	manager	3141	4074	1
8	8	800	manager	3056	4871	2
9	9	750	manager	2528	4907	3
10	10	450	manager	3122	3552	4
11	11	700	manager	3190	3813	5
12	12	500	manager	2790	4601	6

-- Inserare date CENTRE LOGISTICE

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (1, 'Rozei', 89, 59960, 1, 2);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (2, 'Mangnoliei', 61, 76300, 2, 5);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (3, 'Ienuparului', 55, 18579, 3, 2);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (4, 'Carierei', 62, 43393, 4, 6);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (5, 'Sinaia', 80, 98414, 5, 6);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (6, 'Hotelului', 68, 99122, 6, 5);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (7, 'Animalelor', 47, 92302, 1, 2);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (8, 'Calea Mare', 16, 93396, 2, 1);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (9, 'Principala', 76, 93375, 3, 2);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (10, 'Universitatii', 54, 62905, 4, 2);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)  
VALUES (11, 'Dealului', 39, 60850, 5, 1);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,  
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)
```

```
VALUES (12, 'Urcusului', 56, 62170, 6, 5);
```

```
INSERT INTO CENTRE_LOGISTICE (COD_LOGISTIC, STR_LOGI, NR_LOGI,
SUPRAFATA_DEPOZIT, COD_COMPANIE, COD_LOCALITATE)
VALUES (13, 'Muzeului', 66, 61136, 6, 5);
```

	COD_LOGISTIC	STR_LOGI	NR_LOGI	SUPRAFATA_DEPOZIT	COD_COMPANIE	COD_LOCALITATE
1	1	Rozei	89	59960	1	2
2	2	Mangnoliei	61	76300	2	5
3	3	Ienuparului	55	18579	3	2
4	4	Carieriei	62	43393	4	6
5	5	Sinaia	80	98414	5	6
6	6	Hotelului	68	99122	6	5
7	7	Animalelor	47	92302	1	2
8	8	Calea Mare	16	93396	2	1
9	9	Principala	76	93375	3	2
10	10	Universitatii	54	62905	4	2
11	11	Dealului	39	60850	5	1
12	12	Urcusului	56	62170	6	5
13	13	Muzeului	66	61136	6	5

-- Inserare date SPATII COMERCIALE

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (1, 'Strada Mare', 96, 5325, 23452, to_date('11.11.2017',
'DD/MM/YYYY'), 2, 1, 1);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (2, 'Unirii', 161, 7839, 343434, to_date('12.06.2020',
'DD/MM/YYYY'), 2, 2, 2);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (3, 'Revolutiei', 119, 8852, 1132344, to_date('12.11.2019',
'DD/MM/YYYY'), 1, 3, 3);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
```



```
COD_LOCALITATE, COD_LOGISTIC)
VALUES (4, 'Victoriei', 183, 8887, 3434344, to_date('17.11.2021',
'DD/MM/YYYY'), 1, 4, 4);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (5, 'Geamana', 142, 10222, 234343, to_date('05.10.2012',
'DD/MM/YYYY'), 5, 5, 5);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (6, 'Daneza', 171, 9921, 22323, to_date('23.11.2012',
'DD/MM/YYYY'), 4, 6, 6);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (7, 'Margaretei', 33, 5095, 43434, to_date('08.01.2013',
'DD/MM/YYYY'), 3, 1, 7);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (8, 'Rusiei', 200, 12264, 34344, to_date('19.01.2017',
'DD/MM/YYYY'), 2, 2, 8);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (9, 'Germaniei', 95, 9019, 22343, to_date('23.12.2012',
'DD/MM/YYYY'), 2, 3, 9);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (10, 'Polona', 122, 12439, 334344, to_date('01.02.2012',
'DD/MM/YYYY'), 5, 4, 1);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,
COD_LOCALITATE, COD_LOGISTIC)
VALUES (11, 'Mosilor', 196, 8196, 333341, to_date('13.12.2020',
'DD/MM/YYYY'), 6, 5, 2);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,
```

```
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (12, 'Franceza', 46, 9649, 3434343, to_date('22.07.2014',  
'DD/MM/YYYY'), 4, 6, 3);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (13, 'Platanilor', 167, 8630, 324243, to_date('31.07.2016',  
'DD/MM/YYYY'), 3, 1, 4);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (14, 'Horei', 13, 14467, 3434344, to_date('11.02.2017',  
'DD/MM/YYYY'), 1, 2, 5);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (15, 'Prieteniei', 138, 7330, 343433, to_date('11.11.2019',  
'DD/MM/YYYY'), 1, 2, 7);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (16, 'Varsovia', 10, 7800, 503433, to_date('11.10.2019',  
'DD/MM/YYYY'), 1, 1, 1);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (17, 'Calea Domneasca', 101, 7900, 553433, to_date('11.10.2015',  
'DD/MM/YYYY'), 1, 1, 7);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (18, 'Bulevardul Garii', 181, 6900, 53433, to_date('11.10.2017',  
'DD/MM/YYYY'), 1, 8, 7);
```

```
INSERT INTO SPATII_COMERCIALE (COD_MAGAZIN, STR_MAG, NR_MAG,  
SUPRAFATA_MAG, CIFRA_VANZARI, DATA_DESCHIDERE, COD_COMPANIE,  
COD_LOCALITATE, COD_LOGISTIC)  
VALUES (19, 'Protopopescu', 11, 5000, 63433, to_date('21.10.2016',  
'DD/MM/YYYY'), 1, 7, 1);
```

Welcome Page * ProjectsSGBD * SPATII_COMERCIALE *									
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies   Details   Partitions   Indexes   SQL									
Sort... Filter:									
	COD_MAGAZIN	STR_MAG	NR_MAG	SUPRAFATA_MAG	CIFRA_VANZARI	DATA_DESCHIDERE	COD_COMPANIE	COD_LOCALITATE	COD_LOGISTIC
1		1 Strada Mare	96	5325	23452	11-NOV-17	2	1	1
2		2 Unirii	161	7839	343434	12-JUN-20	2	2	2
3		3 Revolutiei	119	8852	1132344	12-NOV-19	1	3	3
4		4 Victoriei	183	8887	3434344	17-NOV-21	1	4	4
5		5 Geamana	142	10222	234343	05-OCT-12	5	5	5
6		6 Daneza	171	9921	22323	23-NOV-12	4	6	6
7		7 Margaretei	33	5095	43434	08-JAN-13	3	1	7
8		8 Rusiei	200	12264	34344	19-JAN-17	2	2	8
9		9 Germaniei	95	9019	22343	23-DEC-12	2	3	9
10		10 Polona	122	12439	334344	01-FEB-12	5	4	1
11		11 Mosilor	196	8196	333341	13-DEC-20	6	5	2
12		12 Franceza	46	9649	3434343	22-JUL-14	4	6	3
13		13 Platanilor	167	8630	324243	31-JUL-16	3	1	4
14		14 Horei	13	14467	3434344	11-FEB-17	1	2	5
15		15 Prieteniei	138	7330	343433	11-NOV-19	1	2	7
16		16 Varsovia	10	7800	503433	11-OCT-19	1	1	1
17		17 Calea Domneasca	101	7900	553433	11-OCT-15	1	1	7
18		18 Bulevardul Garii	181	6900	53433	11-OCT-17	1	8	7
19		19 Protopopescu	11	5000	63433	21-OCT-16	1	7	1

-- Inserare date PRODUCATORI

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (1, 'Romania', 'Fructe legume SRL', 'fructe', 25000, 20);
```

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (2, 'Germania', 'Household', 'parchet' , 65000, 35);
```

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (3, 'Ucraina', 'Cabluri ', 'cabluri' , 10000, 14);
```

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (4, 'Bulgaria', 'Plodviv ', 'carne de porc' , 75000, 22);
```

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (5, 'Franta', 'Baguette', 'branzeturi' , 100000, 40);
```

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (6, 'Cehia', 'Nazdrovia', 'bere' , 25000, 5);
```

```
INSERT INTO PRODUCATORI (COD_PRODUCATOR, TARA, NUME_PRODUCATOR,
NUME_PRODUS, CANTITATE_DISPONIBILA, PRET_UNITAR)
VALUES (7, 'Italia', 'Italiano', 'paste' , 54000, 15);
```

Welcome Page   ProjectSGBD   PRODUCATORI						
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies   Details   Partitions   Indexes   SQL						
Sort..   Filter:						
	COD_PRODUCATOR	TARA	NUME_PRODUCATOR	NUME_PRODUS	CANTITATE_DISPONIBILA	PRET_UNITAR
1	1	Romania	Fructe legume SRL	fructe	25000	20
2	2	Germania	Household	parchet	65000	35
3	3	Ucraina	Cabluri	cabluri	10000	14
4	4	Bulgaria	Plodviv	carne de porc	75000	22
5	5	Franta	Baguette	branzeturi	100000	40
6	6	Cehia	Nazdrovia	bere	250000	5
7	7	Italia	Italiano	paste	540000	15

-- Inserare date DISTRIBUTORI

```
INSERT INTO DISTRIBUTOR (COD_DISTRIBUTOR, CANTITATE_MARFA,
NR_VEHICULE, NUME_DISTRIBUTOR)
VALUES (1, 235000, 120, 'Fulger Transport');
```

```
INSERT INTO DISTRIBUTOR (COD_DISTRIBUTOR, CANTITATE_MARFA,
NR_VEHICULE, NUME_DISTRIBUTOR)
VALUES (2, 39000, 20, 'DVL');
```

```
INSERT INTO DISTRIBUTOR (COD_DISTRIBUTOR, CANTITATE_MARFA,
NR_VEHICULE, NUME_DISTRIBUTOR)
VALUES (3, 500000, 100, 'Transport Rapid SRL');
```

```
INSERT INTO DISTRIBUTOR (COD_DISTRIBUTOR, CANTITATE_MARFA,
NR_VEHICULE, NUME_DISTRIBUTOR)
VALUES (4, 9000, 20, 'ROG');
```

```
INSERT INTO DISTRIBUTOR (COD_DISTRIBUTOR, CANTITATE_MARFA,
NR_VEHICULE, NUME_DISTRIBUTOR)
VALUES (5, 700000, 160, 'Cargurs Marfa');
```

Welcome Page   ProjectSGBD   DISTRIBUTOR				
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies   Details   Partitions				
Sort..   Filter:				
	COD_DISTRIBUTOR	CANTITATE_MARFA	NR_VEHICULE	NUME_DISTRIBUTOR
1	1	235000	120	Fulger Transport
2	2	39000	20	DVL
3	3	500000	100	Transport Rapid SRL
4	4	9000	20	ROG
5	5	700000	160	Cargurs Marfa

-- Inserare date CONTRACTE

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (1, 11232, 2, 1, 1, to_date('21.10.2020', 'DD/MM/YYYY'),  
to_date('21.10.2023', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (2, 4343, 5, 2, 2, to_date('01.01.2019', 'DD/MM/YYYY'),  
to_date('01.01.2019', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (3, 2322, 5, 3, 3, to_date('18.10.2018', 'DD/MM/YYYY'),  
to_date('18.10.2021', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (4, 34555, 5, 4, 4, to_date('15.07.2021', 'DD/MM/YYYY'),  
to_date('15.07.2024', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (5, 23444, 5, 5, 5, to_date('20.10.2022', 'DD/MM/YYYY'),  
to_date('21.10.2025', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (6, 45454, 3, 1, 6, to_date('20.02.2021', 'DD/MM/YYYY'),  
to_date('22.02.2023', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (7, 234443, 3, 2, 7, to_date('01.10.2019', 'DD/MM/YYYY'),  
to_date('31.12.2025', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (8, 45344, 3, 3, 8, to_date('25.05.2020', 'DD/MM/YYYY'),  
to_date('25.05.2018', 'DD/MM/YYYY'));
```

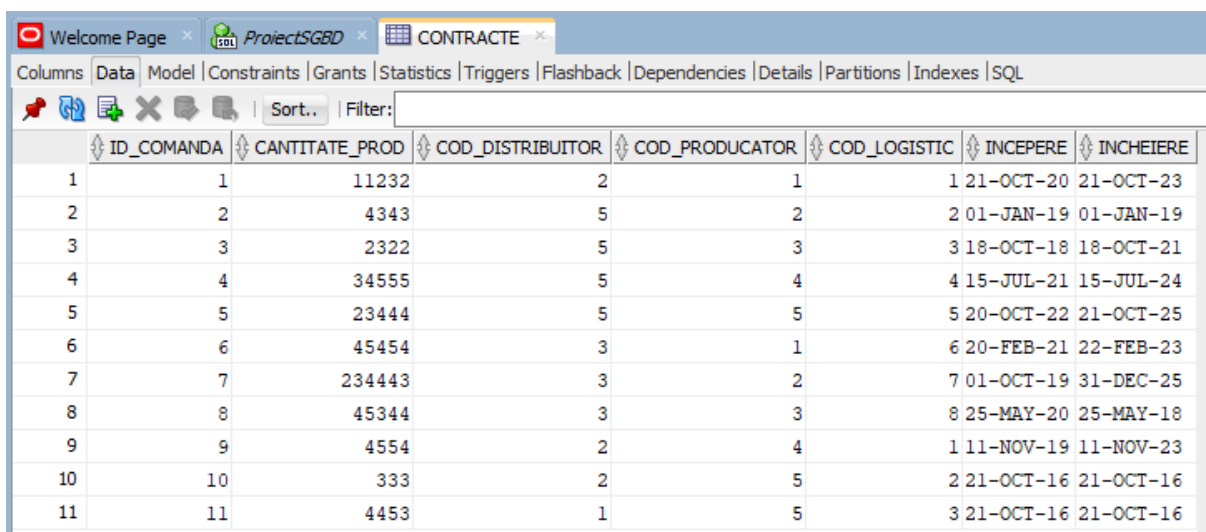
```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (9, 4554, 2, 4, 1, to_date('11.11.2019', 'DD/MM/YYYY'),
```

```
to_date('11.11.2023', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUTOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (10, 333, 2, 5, 2, to_date('21.10.2016', 'DD/MM/YYYY'),  
to_date('21.10.2016', 'DD/MM/YYYY'));
```

```
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUTOR,  
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)  
VALUES (11, 4453, 1, 5, 3, to_date('21.10.2016', 'DD/MM/YYYY'),  
to_date('21.10.2016', 'DD/MM/YYYY'));
```

```
COMMIT;
```



	ID_COMANDA	CANTITATE_PROD	COD_DISTRIBUTOR	COD_PRODUCATOR	COD_LOGISTIC	INCEPERE	INCHEIERE
1	1	11232	2	1	1	21-OCT-20	21-OCT-23
2	2	4343	5	2	2	01-JAN-19	01-JAN-19
3	3	2322	5	3	3	18-OCT-18	18-OCT-21
4	4	34555	5	4	4	15-JUL-21	15-JUL-24
5	5	23444	5	5	5	20-OCT-22	21-OCT-25
6	6	45454	3	1	6	20-FEB-21	22-FEB-23
7	7	234443	3	2	7	01-OCT-19	31-DEC-25
8	8	45344	3	3	8	25-MAY-20	25-MAY-18
9	9	4554	2	4	1	11-NOV-19	11-NOV-23
10	10	333	2	5	2	21-OCT-16	21-OCT-16
11	11	4453	1	5	3	21-OCT-16	21-OCT-16

## 6. Rezolvare exerciții

### 6.1 Exercițiul 6

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze două tipuri de colecție studiate. Apelați subprogramul.

```
-- <<Exercitiul 6>>
```

```
-- Sa se analizeze situatia financiara a unui anumit tip de magazine dintr-un  
judet (ambele date de utilizator).
```

```
-- Pentru fiecare magazin in parte se va prezenta cat la suta reprezinta din  
cifra totala de afaceri a companiei si detalii generale privind anul deschiderii  
si suprafata acestuia.
```

```

CREATE OR REPLACE PROCEDURE ex_6
(v_tip_magazin IN lant_companii.tip_magazin%TYPE, v_num_judet IN
judete.num_jud%TYPE)
IS
    TYPE rec_magazin IS RECORD
    ( s_mag spatii_comerciale.suprafata_mag%TYPE,
      ca spatii_comerciale.cifra_vanzari%TYPE,
      localitate spatii_comerciale.cod_localitate%TYPE,
      companie spatii_comerciale.cod_companie%TYPE,
      an spatii_comerciale.data_deschidere%TYPE,
      cod spatii_comerciale.cod_magazin%TYPE
    );

    TYPE tab_magazine IS TABLE OF rec_magazin INDEX BY BINARY_INTEGER;
    t_mag_profil tab_magazine;

    -- tabel imbricat
    TYPE tab_companii IS TABLE OF lant_companii.cod_companie%TYPE;
    t_comp_profil tab_companii;

    TYPE tab_localitati IS TABLE OF localitati.cod_localitate%TYPE;
    t_loc tab_localitati;

    v_cod_judet judete.cod_judet%TYPE;
    v_comp lant_companii.num_companie%TYPE;
    v_loc localitati.num_loc%TYPE;
    v_ca_totala lant_companii.cifra_afaceri%TYPE;

BEGIN
    BEGIN -- verificarea datelor introduce
        SELECT cod_judet
        INTO v_cod_judet
        FROM judete
        WHERE lower(num_jud) = lower(v_num_judet);
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            v_cod_judet := NULL;
            DBMS_OUTPUT.PUT_LINE (upper(v_num_judet)||' nu este un judet!');
    END;
    IF v_cod_judet IS NOT NULL THEN
        SELECT cod_companie
        BULK COLLECT INTO t_comp_profil
        FROM lant_companii
        WHERE lower(tip_magazin) = lower(v_tip_magazin);

        IF t_comp_profil.COUNT = 0 THEN -- verificarea datelor introduce
            DBMS_OUTPUT.PUT_LINE ('Acest tip de magazin:
'||upper(v_tip_magazin)||', nu este inregistrat in baza de date!');

```

```

ELSE
    DBMS_OUTPUT.PUT_LINE ('<< Situatia magazinelor tip ' ||
upper(v_tip_magazin) || ' din judetul ' || upper(v_ume_judet) || '>>');
    SELECT cod_localitate
    BULK COLLECT INTO t_loc
    FROM localitati
    WHERE v_cod_judet = cod_judet;

    SELECT suprafata_mag, cifra_vanzari, cod_localitate, cod_comanie,
data_deschidere, cod_magazin
    BULK COLLECT INTO t_mag_profil
    FROM spatii_comerciale
    ORDER BY cod_comanie, cod_localitate, cifra_vanzari, suprafata_mag;

    FOR i IN t_mag_profil.FIRST..t_mag_profil.LAST LOOP
        IF t_mag_profil(i).comanie MEMBER OF t_comp_profil AND
t_mag_profil(i).localitate MEMBER OF t_loc THEN

            SELECT nume_comanie, cifra_afaceri
            INTO v_comp, v_ca_totala
            FROM lant_comanii
            WHERE cod_comanie = t_mag_profil(i).comanie;

            SELECT nume_loc
            INTO v_loc
            FROM localitati
            WHERE cod_localitate = t_mag_profil(i).localitate;

            DBMS_OUTPUT.PUT_LINE('<O> ' || upper(v_comp) || ' ' ||
t_mag_profil(i).cod || ' ' || v_loc);
            DBMS_OUTPUT.PUT_LINE(' # Procent cira afaceri: ' ||
trunc(t_mag_profil(i).ca/ v_ca_totala, 3)|| '%');
            DBMS_OUTPUT.PUT_LINE(' # An deschidere: ' ||
t_mag_profil(i).an);
            DBMS_OUTPUT.PUT_LINE(' # Suprafata: ' ||
t_mag_profil(i).s_mag || ' m.p. ');
        END IF;
    END LOOP;

    END IF;
END IF;
END ex_6;
/
BEGIN
    ex_6('discount', 'dâmbovita');
END;
/

```



```

Procedure EX_6 compiled

<< Situatiia magazinelor tip DISCOUNT din judetul DÂMBOVITA>>
<O> KAUF LAND 16 Targoviste
# Procent cira afaceri: .419%
# An deschidere: 11-OCT-19
# Suprafata: 7800 m.p.
<O> KAUF LAND 17 Targoviste
# Procent cira afaceri: .461%
# An deschidere: 11-OCT-15
# Suprafata: 7900 m.p.
<O> KAUF LAND 3 Moreni
# Procent cira afaceri: .943%
# An deschidere: 12-NOV-19
# Suprafata: 8852 m.p.
<O> KAUF LAND 4 Pucioasa
# Procent cira afaceri: 2.861%
# An deschidere: 17-NOV-21
# Suprafata: 8887 m.p.
<O> KAUF LAND 19 Gaesti
# Procent cira afaceri: .052%
# An deschidere: 21-OCT-16
# Suprafata: 5000 m.p.
<O> KAUF LAND 18 Titu
# Procent cira afaceri: .044%
# An deschidere: 11-OCT-17
# Suprafata: 6900 m.p.
<O> LIDL 1 Targoviste
# Procent cira afaceri: .023%
# An deschidere: 11-NOV-17
# Suprafata: 5325 m.p.
<O> LIDL 9 Moreni
# Procent cira afaceri: .022%
# An deschidere: 23-DEC-12
# Suprafata: 9019 m.p.

```

## 6.2 Exercițiul 7

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze 2 tipuri de cursoare studiate, unul dintre acestea fiind cursor parametrizat. Apelați subprogramul.

```
-- <<Exercitiul 7>>
```

```

--      Sa se afiseze pentru magazinele de un anumit tip (discount, constructii,
proximitate) situatia actuala
-- a aprovizionarii. Se va calcula pretul aprovizionarii fiecarui magazin,
stiind ca marfa de la un centru logistic este impartita
-- in mod egal intre magazine.

```

```

CREATE OR REPLACE PROCEDURE ex_7 (v_tip_magazin IN
lant_companii.tip_magazin%TYPE) IS
    CURSOR magazine IS
        SELECT cod_magazin, lc.cod_companie, nume_companie, nume_loc,
cod_logistic
        FROM spatii_comerciale sc, lant_companii lc, localitati l
        WHERE sc.cod_companie = lc.cod_companie
            AND sc.cod_localitate = l.cod_localitate
        ORDER BY nume_companie, l.nume_loc;

    CURSOR aprovizionare (centru_logistic contracte.cod_logistic%TYPE) IS
        SELECT nume_producator, nume_produs, tara, cantitate_prod,
nume_distribuator, p.pret_unitar, c.id_comanda
        FROM contracte c, distribuitor d, producatori p
        WHERE c.cod_producator = p.cod_producator
            AND c.cod_distribuitor = d.cod_distribuitor
            AND c.cod_logistic = centru_logistic;

v_cod_magazin spatii_comerciale.cod_magazin%TYPE;
v_cod_companie spatii_comerciale.cod_companie%TYPE;
v_nume_companie lant_companii.nume_companie%TYPE;
v_nume_loc localitati.nume_loc%TYPE;
v_cod_logistic spatii_comerciale.cod_logistic%TYPE;
v_aux INTEGER;
TYPE tab_mag_tip IS TABLE OF lant_companii.cod_companie%TYPE;
mag_tip tab_mag_tip;

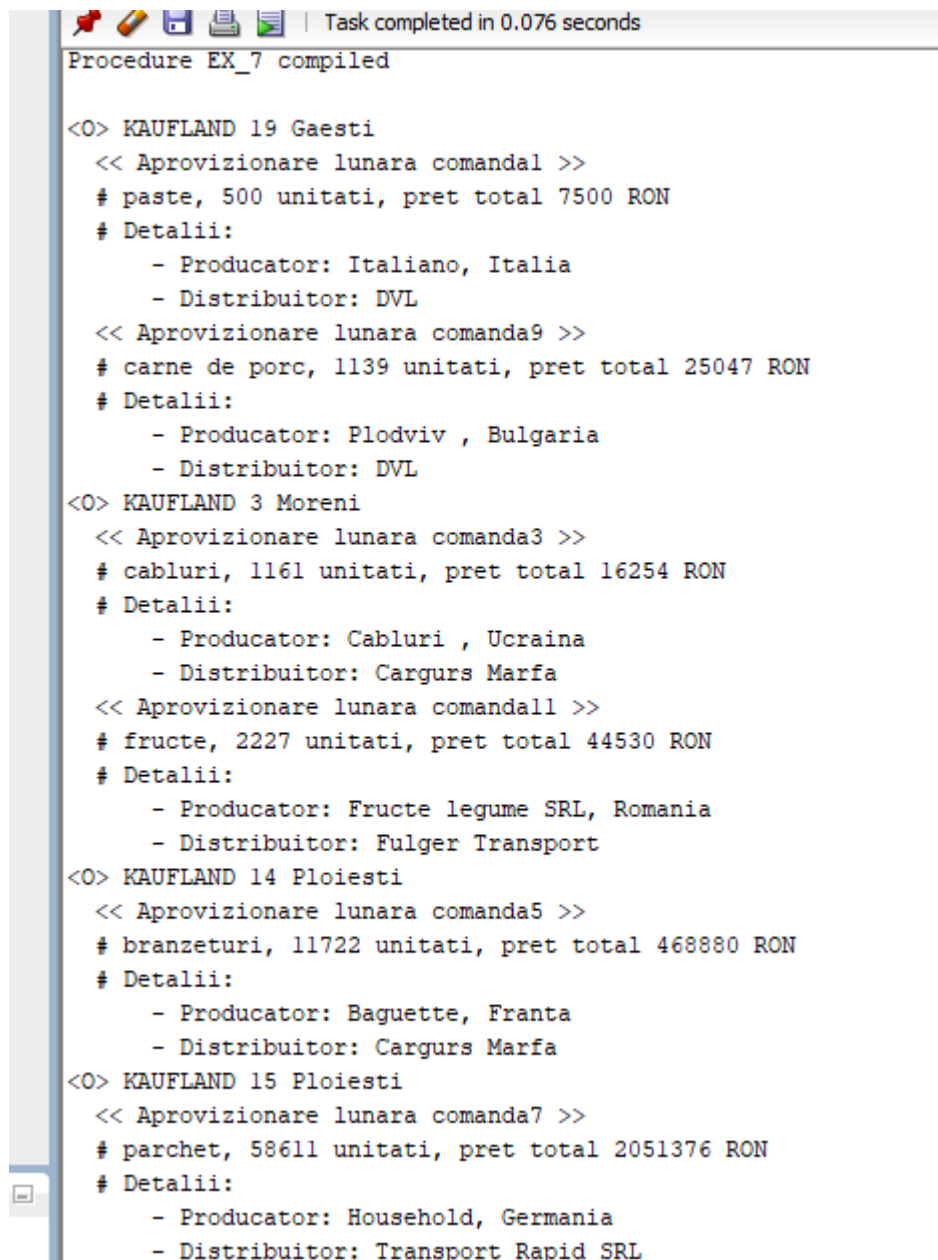
BEGIN
    -- se inregistreaza companiile care sunt de tipul introdus
    SELECT cod_companie BULK COLLECT INTO mag_tip
    FROM lant_companii
    WHERE LOWER(v_tip_magazin) = LOWER(tip_magazin);
    -- DBMS_OUTPUT.PUT_LINE(mag_tip.COUNT);
    OPEN magazine;
    LOOP
        FETCH magazine INTO v_cod_magazin, v_cod_companie, v_nume_companie,
v_nume_loc, v_cod_logistic;
        EXIT WHEN magazine%NOTFOUND;
        -- DBMS_OUTPUT.PUT_LINE(v_cod_companie);
        -- pentru magazinele care sunt de tipul introdus se afiseaza
aprovizionarea
        IF v_cod_companie MEMBER OF mag_tip THEN
            DBMS_OUTPUT.PUT_LINE('<O> ' || upper(v_nume_companie) || ' ' ||
v_cod_magazin || ' ' || v_nume_loc);
            FOR a IN aprovizionare (v_cod_logistic) LOOP
                -- se calculeaza numarul de magazine care se aprovizioneaza
de la acelasi
                -- centru logistic
                SELECT COUNT(v_cod_magazin) INTO v_aux FROM
spatii_comerciale WHERE cod_logistic = v_cod_logistic;
                DBMS_OUTPUT.PUT_LINE ( ' ' || '<< Aprovizionare lunara

```

```

comanda' || a.id_comanda||' >>');
        DBMS_OUTPUT.PUT_LINE (' # '||a.ume_produs||', ' ||
ROUND(a.cantitate_prod/v_aux) || ' unitati, pret total ' ||
ROUND(a.cantitate_prod/v_aux*a.pret_unitar) || ' RON');
        DBMS_OUTPUT.PUT_LINE (' # Detalii:');
        DBMS_OUTPUT.PUT_LINE('      - Producator: ' ||
a.ume_producator || ', ' || a.tara);
        DBMS_OUTPUT.PUT_LINE('      - Distribuitor: ' ||
a.ume_distribuitor);
        END LOOP;
    END IF;
END LOOP;
CLOSE magazine;
END ex_7;
/

```



```

Task completed in 0.076 seconds
Procedure EX_7 compiled

<O> KAUF LAND 19 Gaesti
  << Aprovizionare lunara comandal >>
  # paste, 500 unitati, pret total 7500 RON
  # Detalii:
    - Producator: Italiano, Italia
    - Distribuitor: DVL
  << Aprovizionare lunara comanda9 >>
  # carne de porc, 1139 unitati, pret total 25047 RON
  # Detalii:
    - Producator: Plodviv , Bulgaria
    - Distribuitor: DVL
<O> KAUF LAND 3 Moreni
  << Aprovizionare lunara comanda3 >>
  # cabluri, 1161 unitati, pret total 16254 RON
  # Detalii:
    - Producator: Cabluri , Ucraina
    - Distribuitor: Cargurs Marfa
  << Aprovizionare lunara comandall >>
  # fructe, 2227 unitati, pret total 44530 RON
  # Detalii:
    - Producator: Fructe legume SRL, Romania
    - Distribuitor: Fulger Transport
<O> KAUF LAND 14 Ploiesti
  << Aprovizionare lunara comanda5 >>
  # branzeturi, 11722 unitati, pret total 468880 RON
  # Detalii:
    - Producator: Baguette, Franta
    - Distribuitor: Cargurs Marfa
<O> KAUF LAND 15 Ploiesti
  << Aprovizionare lunara comanda7 >>
  # parchet, 58611 unitati, pret total 2051376 RON
  # Detalii:
    - Producator: Household, Germania
    - Distribuitor: Transport Rapid SRL

```

### 6.3 Exercițiul 8

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite. Definiți minim 2 excepții. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

```
-- <<Exercitiul 8>>
```

```
--      Sa se afiseze pentru o companie x pretul mediu platit per minut pentru publicitate televizata.
```

```
CREATE OR REPLACE FUNCTION ex_8(v_nume_companie IN
lant_companii.nume_companie%TYPE)
RETURN NUMBER IS v_pret_mediu FLOAT := NULL;
    v_cod_companie lant_companii.cod_companie%TYPE;
    exc_fara_televizat EXCEPTION;
BEGIN
    SELECT cod_companie INTO v_cod_companie
    FROM lant_companii
    WHERE lower(v_nume_companie) = lower(nume_companie);

    -- calculez pretul mediu in functie de toate contractele televizate
    aflate in desfasurare
    BEGIN
        SELECT ROUND(SUM(t.difuzare_int1*60/t.durata_spot*pt.cost_int1 +
t.difuzare_int2*60/t.durata_spot*pt.cost_int2) /
                    SUM(t.difuzare_int1 + t.difuzare_int2))
        INTO v_pret_mediu
        FROM televizat t, post_tv pt, campanii_marketing cm
        WHERE t.cod_tv = pt.cod_tv
            AND t.cod_marketing = cm.cod_marketing
            AND cm.cod_companie = v_cod_companie
            AND cm.finalizare - SYSDATE >0
        GROUP BY cm.cod_companie;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            RAISE_APPLICATION_ERROR(-20080, 'Nu exista contracte televizate
in desfasurare!');
    END;
    RETURN v_pret_mediu;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            RAISE_APPLICATION_ERROR(-20081, 'Nu exista magazinul introdus!');
END ex_8;
/
-- Caz 'Nu exista magazinul introdus!'
BEGIN
    DBMS_OUTPUT.PUT_LINE (ex_8('Carrefour'));
END;
```

```

/
-- Caz 'Nu exista contracte televizate in desfasurare!'
BEGIN
    DBMS_OUTPUT.PUT_LINE (ex_8('Lidl'));
END;
/
-- Caz acceptat
BEGIN
    DBMS_OUTPUT.PUT_LINE (ex_8('Kaufland'));
END;
/

```

```

Error starting at line : 199 in command -
BEGIN
    DBMS_OUTPUT.PUT_LINE (ex_8('Carrefour'));
END;
Error report -
ORA-20081: Nu exista magazinul introdus!
ORA-06512: at "ANDITOADER.EX_8", line 28
ORA-06512: at line 2

Error starting at line : 204 in command -
BEGIN
    DBMS_OUTPUT.PUT_LINE (ex_8('Lidl'));
END;
Error report -
ORA-20080: Nu exista contracte televizate in desfasurare!
ORA-06512: at "ANDITOADER.EX_8", line 23
ORA-06512: at line 2

986 RON/minut

```

## 6.4 Exercițiul 9

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip procedură care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO\_DATA\_FOUND și TOO\_MANY\_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

```
-- <<Exercitiul 9>>
```

```
--      Sa se afiseze pentru cel mai performant magazin X din fiecare localitate
--      (cea mai mare cifra de afaceri) informatii
--      legate de campaniile offline desfasurate de companie.
```

```
CREATE OR REPLACE FUNCTION cel_mai_bun(v_cod_companie IN
spatii_comerciale.cod_companie%TYPE, v_cod_localitate IN
spatii_comerciale.cod_localitate%TYPE)
```

```

RETURN spatii_comerciale.cod_magazin%TYPE IS
v_cod spatii_comerciale.cod_magazin%TYPE := -1;
BEGIN
    SELECT cod_magazin INTO v_cod
    FROM spatii_comerciale
    WHERE cifra_vanzari =(
        SELECT MAX(cifra_vanzari)
        FROM spatii_comerciale
        WHERE cod_companie = v_cod_companie
        AND cod_localitate = v_cod_localitate);

    RETURN v_cod;
END cel_mai_bun;
/

CREATE OR REPLACE PROCEDURE ex_9
(v_nume_companie IN lant_companii.nume_companie%TYPE)
IS
    v_cod_companie lant_companii.cod_companie%TYPE;
    v_cod_loc localitati.cod_localitate%TYPE;
    v_cod_mag_1 spatii_comerciale.cod_magazin%TYPE;
    v_ca spatii_comerciale.cifra_vanzari%TYPE;
    TYPE rec_marketing IS RECORD(
        nume campanii_marketing.nume_companie%TYPE,
        nr_rev localitati_tinta.nr_reviste%TYPE,
        nr_pli localitati_tinta.nr_pliante%TYPE,
        nr_pan localitati_tinta.nr_panouri%TYPE
    );
    TYPE tab_marketing IS TABLE OF rec_marketing INDEX BY PLS_INTEGER;
    info_marketing tab_marketing;
    -- selectez localitatile care au magazin(e) de la compania introdusa
    CURSOR loc_mag_comp (v_cod_companie spatii_comerciale.cod_companie%TYPE) IS
        SELECT DISTINCT sc.cod_localitate, l.nume_loc, j.nume_jud
        FROM spatii_comerciale sc, localitati l, judete j
        WHERE cod_companie = v_cod_companie
            AND sc.cod_localitate = l.cod_localitate
            AND l.cod_judet = j.cod_judet
        ORDER BY j.nume_jud, l.nume_loc;
    -- selectez detalii despre campaniile de marketing desfasurate de compania X
    in toate localitatile
    CURSOR info_campanii (v_cod_companie spatii_comerciale.cod_companie%TYPE) IS
        SELECT lt.cod_localitate, cm.nume_companie, nvl(lt.nr_reviste, 0) as
rev, nvl(lt.nr_pliante, 0) as pli, nvl(lt.nr_panouri, 0) as pan
        FROM localitati_tinta lt, camp_offline co, campanii_marketing cm
        WHERE lt.cod_offline = co.cod_offline
            AND co.cod_marketing = cm.cod_marketing
            AND cm.cod_companie = v_cod_companie
            AND cm.finalizare - SYSDATE > 0;
BEGIN
    SELECT cod_companie INTO v_cod_companie
    FROM lant_companii
    WHERE lower(v_nume_companie) = lower(nume_companie);

```

```

-- salvez informatiile despre campaniile aflate in desfasurare de compani X
in toate localitatile
FOR mkt IN info_campanii(v_cod_companie) LOOP
    info_marketing(mkt.cod_localitate).nume := mkt.nume_companie;
    info_marketing(mkt.cod_localitate).nr_rev := mkt.rev;
    info_marketing(mkt.cod_localitate).nr_pli := mkt.pli;
    info_marketing(mkt.cod_localitate).nr_pan := mkt.pan;
END LOOP;

FOR loc IN loc_mag_comp(v_cod_companie) LOOP
    v_cod_mag_1 := cel_mai_bun(v_cod_companie, loc.cod_localitate);
    SELECT cifra_vanzari INTO v_ca FROM spatii_comerciale WHERE cod_magazin
= v_cod_mag_1;
    DBMS_OUTPUT.PUT_LINE('<O>' || ' ' || upper(v_nume_companie) || ' ' ||
v_cod_mag_1 || ' ' || loc.nume_jud || ' ' || loc.nume_loc);
    DBMS_OUTPUT.PUT_LINE(' # Cifra vanzari: ' || v_ca || ' RON/luna');
    -- afisez info despre campanii de marketing, daca ea exista
    BEGIN
        DBMS_OUTPUT.PUT_LINE(' # Campania:
' || upper(info_marketing(loc.cod_localitate).nume));
        DBMS_OUTPUT.PUT_LINE(' - Numar reviste: ' ||
info_marketing(loc.cod_localitate).nr_rev);
        DBMS_OUTPUT.PUT_LINE(' - Numar pliante: ' ||
info_marketing(loc.cod_localitate).nr_pli);
        DBMS_OUTPUT.PUT_LINE(' - Numar panouri publicitare: ' ||
info_marketing(loc.cod_localitate).nr_pan);
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            DBMS_OUTPUT.PUT_LINE(' # NU EXISTA CAMPANIE DE MARKETING AFLATA IN
DESFASURARE! ');
        END;
    END LOOP;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20090, 'Magazinul introdus nu a fost
gasit!');
    WHEN TOO_MANY_ROWS THEN
        RAISE_APPLICATION_ERROR(-20091, 'Exista mai multe magazine cu
aceeasi cifra de vanzari!');
    WHEN CURSOR_ALREADY_OPEN THEN
        RAISE_APPLICATION_ERROR(-20092, 'Accesare multipla a aceeasi zona de
memorie!');
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20093, 'Alta eroare!');
END ex_9;
/
-- Caz 'Magazinul introdus nu a fost gasit!'
BEGIN
    ex_9('carrefour');
END;

```

```

/
-- Caz 'Exista mai multe magazine cu aceeasi cifra de vanzari!'
BEGIN
    ex_9('profi');
END;
/
-- Caz acceptare
BEGIN
    ex_9('kaufland');
END;
/

```

Procedure EX\_9 compiled

Error starting at line : 306 in command -  
BEGIN

```

    ex_9('carrefour');
END;

```

Error report -

```

ORA-20090: Magazinul introdus nu a fost gasit!
ORA-06512: at "ANDITOADER.EX_9", line 63
ORA-06512: at line 2

```

Error starting at line : 311 in command -  
BEGIN

```

    ex_9('profi');
END;

```

Error report -

```

ORA-20091: Exista mai multe magazine cu aceeasi cifra de vanzari!
ORA-06512: at "ANDITOADER.EX_9", line 65
ORA-06512: at line 2

```

<O> KAUF LAND 19 Dâmbovita Gaesti

# Cifra vanzari: 63433 RON/luna

# NU EXISTA CAMPANIE DE MARKETING AFLATA IN DESFASURARE!

<O> KAUF LAND 3 Dâmbovita Moreni

# Cifra vanzari: 1132344 RON/luna

# Campania: KAUF LANDTASTIC

- Numar reviste: 3000

- Numar pliante: 0

- Numar panouri publicitare: 23



## 6.5 Exercițiul 10

Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.

```
--      Sa se realizeze un trigger de tip LMD la nivel de comanda care sa
verifice ca numarul de distribuitori
-- nu reprezinta mai multe de un sfert din cel al producatorilor (pentru a nu se
satura piata trasnportatorilor).

CREATE OR REPLACE TRIGGER ex_10
  BEFORE INSERT ON distribuitor
DECLARE
  pragma autonomous_transaction;
  v_nr_dis INT;
  v_nr_prod INT;
BEGIN
  SELECT COUNT(cod_distribuitor) INTO v_nr_dis FROM distribuitor;
  SELECT COUNT(cod_producator) INTO v_nr_prod FROM producatori;
  IF v_nr_dis > v_nr_prod/4 THEN
    RAISE_APPLICATION_ERROR(-20100,'S-A ATINS NUMARUL MAXIM DE DISTRIBUITORI
NECESARI!');
  END IF;
END ex_10;
/

-- Declansare TRIGGER
BEGIN
  FOR i IN 1 .. 5 LOOP
    INSERT INTO distribuitor VALUES (i*100, 700000, 160, 'Cargurs Marfa');
  END LOOP;
END;
/

-- Sterge trigger
DROP TRIGGER ex_10;
/
```

```

22
23 -- Declansare TRIGGER
24 BEGIN
25     FOR i IN 1 .. 5 LOOP
26         INSERT INTO distribuitor VALUES (i*100, 700000, 160, 'Cargurs Marfa');
27     END LOOP;
28 END;
29
30 -- Sterge trigger
31 DROP TRIGGER ex_10;
32 /

```

Script Output x Query Result x  
Task completed in 0.047 seconds

```

Error starting at line : 24 in command -
BEGIN
    FOR i IN 1 .. 5 LOOP
        INSERT INTO distribuitor VALUES (i*100, 700000, 160, 'Cargurs Marfa');
    END LOOP;
END;
Error report -
ORA-20100: S-A ATINS NUMARUL MAXIM DE DISTRIBUTORI NECESARI!
ORA-06512: at "ANDITOADER.EX_10", line 9
ORA-04088: error during execution of trigger 'ANDITOADER.EX_10'
ORA-06512: at line 3

```

## 6.6 Exercițiul 11

Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

```
-- Sa se realizeze un trigger de tip LMD la nivel de linie care sa verifice
cantitatea de produse cerute in functie
-- de cea existenta la producator la introducerea si actualizarea contractelor.
Pentru stergerea de contracte, se va verifica ca acestea nu
-- se mai afla in desfasurare.
```

```
-- functie care verifica disponibilitatea distribuitorilor
CREATE OR REPLACE FUNCTION disponibilitate_distribuitor(v_distribuitor IN
distribuitor.cod_distribuitor%TYPE)
RETURN distribuitor.cantitate_marfa%TYPE IS
liber distribuitor.cantitate_marfa%TYPE := NULL;
BEGIN
    SELECT cantitate_marfa - (SELECT SUM(cantitate_prod)
                                FROM contracte
                                GROUP BY cod_distribuitor
                                HAVING cod_distribuitor = v_distribuitor)
    INTO liber
    FROM distribuitor
    WHERE cod_distribuitor = v_distribuitor;
    -- tratarea cazului cand distribuitorul e liber
    IF liber is NULL THEN
        SELECT cantitate_marfa INTO liber FROM distribuitor WHERE
cod_distribuitor = v_distribuitor;
```

```

        END IF;
        RETURN liber;
END disponibilitate_distribuator;
/

BEGIN
    DBMS_OUTPUT.PUT_LINE(disponibilitate_distribuator(4));
END;
-- functie care verifica disponibilitatea produselor la furnizori
CREATE OR REPLACE FUNCTION stoc_disponibil(producator IN
contracte.cod_producator%TYPE)
RETURN producatori.cantitate_disponibila%TYPE IS
stoc producatori.cantitate_disponibila%TYPE := NULL;
BEGIN
    SELECT cantitate_disponibila - (SELECT SUM(cantitate_prod)
                                   FROM contracte
                                   GROUP BY cod_producator
                                   HAVING cod_producator = producator)

    INTO stoc
    FROM producatori
    WHERE cod_producator = producator;
    -- tratarea cazului cand distribuitorul e liber
    IF stoc is NULL THEN
        SELECT cantitate_disponibila INTO stoc FROM producatori WHERE
cod_producator = producator;
    END IF;
    RETURN stoc;
END stoc_disponibil;
/

BEGIN
    DBMS_OUTPUT.PUT_LINE(stoc_disponibil(1));
END;

CREATE OR REPLACE TRIGGER ex_11
    BEFORE INSERT OR UPDATE OR DELETE ON contracte
    FOR EACH ROW
DECLARE
    pragma autonomous_transaction;
    exceptie_stoc EXCEPTION;
    exceptie_distribuator EXCEPTION;
    exceptie_data EXCEPTION;
BEGIN
    -- se verifica ca marfa&distribuitorul sa fie disponibili
    IF INSERTING THEN -- se verifica ca marfa sa fie disponibila

        IF :NEW.cantitate_prod > stoc_disponibil(:NEW.cod_producator) THEN
            RAISE exceptie_stoc;
        ELSIF :NEW.cantitate_prod >
disponibilitate_distribuator(:NEW.cod_distribuator) THEN
            RAISE exceptie_distribuator;
        END IF;

```

```

-- se verifica ca marfa&distribuitorul sa fie disponibili
    ELSIF UPDATING THEN
        -- cand se schimba producatorul, trebuie sa fie integral disponibilia
        cantitatea contractata
            IF :OLD.cod_producator != :NEW.cod_producator THEN
                IF :NEW.cantitate_prod > stoc_disponibil(:NEW.cod_producator) THEN
                    RAISE exceptie_stoc;
                ELSIF :NEW.cantitate_prod >
disponibilitate_distribuitor(:NEW.cod_distribuitor) THEN
                    RAISE exceptie_distribuitor;
                END IF;
            -- cand se pastreaza acelasi producator, se verifica ca diferenta sa fie
            disponibila
                ELSE
                    IF :NEW.cantitate_prod - :OLD.cantitate_prod >
stoc_disponibil(:NEW.cod_producator) THEN
                        RAISE exceptie_stoc;
                    ELSIF :NEW.cantitate_prod - :OLD.cantitate_prod >
disponibilitate_distribuitor(:NEW.cod_distribuitor) THEN
                        RAISE exceptie_distribuitor;
                    END IF;
                END IF;
            -- cand se sterg contracte aflate in desfasurare
            ELSIF DELETING THEN
                IF :OLD.INCHEIERE - SYSDATE > 0 THEN
                    RAISE exceptie_data;
                END IF;
            END IF;
EXCEPTION
    WHEN exceptie_stoc THEN
        RAISE_APPLICATION_ERROR(-20110,'STOC INDISPONIBIL!');
    WHEN exceptie_distribuitor THEN
        RAISE_APPLICATION_ERROR(-20111,'DISTRIBUTOR OCUPAT!');
    WHEN exceptie_data THEN
        RAISE_APPLICATION_ERROR(-20112,'CONTRACT IN DESFASURARE!');
END ex_11;
/

```

-- Verificare TRIGGER

```

-- Activare INSERT
-- STOC INDISPONIBIL
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUTOR,
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)
VALUES (101, 112000, 2, 1, 1, to_date('21.10.2020', 'DD/MM/YYYY'),
to_date('21.10.2023', 'DD/MM/YYYY'));
-- DISTRIBUTOR OCUPAT
INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUTOR,
COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)
VALUES (102, 50000, 4, 7, 1, to_date('21.10.2020', 'DD/MM/YYYY'),
to_date('21.10.2023', 'DD/MM/YYYY'));

```

```

105 -- Verificare TRIGGER
106
107 -- Activare INSERT
108 -- STOC INDISPONIBIL
109 INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR, COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)
110 VALUES (101, 112000, 2, 1, 1, to_date('21.10.2020', 'DD/MM/YYYY'), to_date('21.10.2023', 'DD/MM/YYYY'));
111 -- DISTRIBUITOR OCUPAT
112 INSERT INTO CONTRACTE (ID_COMANDA, CANTITATE_PROD, COD_DISTRIBUITOR, COD_PRODUCATOR, COD_LOGISTIC, INCEPERE, INCHEIERE)
113 VALUES (102, 50000, 4, 7, 1, to_date('21.10.2020', 'DD/MM/YYYY'), to_date('21.10.2023', 'DD/MM/YYYY'));
114
115 -- Activare UPDATE
116 -- STOC INDISPONIBIL
117 UPDATE contracte SET cantitate_prod = 100000 WHERE id_comanda = 100;
118 -- DISTRIBUITOR OCUPAT

```

Script Output x Query Result x

Task completed in 0.053 seconds

Error starting at line : 109 in command -  
 INSERT INTO CONTRACTE (ID\_COMANDA, CANTITATE\_PROD, COD\_DISTRIBUITOR, COD\_PRODUCATOR, COD\_LOGISTIC, INCEPERE, INCHEIERE)  
 VALUES (101, 112000, 2, 1, 1, to\_date('21.10.2020', 'DD/MM/YYYY'), to\_date('21.10.2023', 'DD/MM/YYYY'))  
 Error report -  
 ORA-20110: STOC INDISPONIBIL!  
 ORA-06512: at "ANDITOADER.EX\_11", line 40  
 ORA-04088: error during execution of trigger 'ANDITOADER.EX\_11'

Error starting at line : 112 in command -  
 INSERT INTO CONTRACTE (ID\_COMANDA, CANTITATE\_PROD, COD\_DISTRIBUITOR, COD\_PRODUCATOR, COD\_LOGISTIC, INCEPERE, INCHEIERE)  
 VALUES (102, 50000, 4, 7, 1, to\_date('21.10.2020', 'DD/MM/YYYY'), to\_date('21.10.2023', 'DD/MM/YYYY'))  
 Error report -  
 ORA-20111: DISTRIBUITOR OCUPAT!  
 ORA-06512: at "ANDITOADER.EX\_11", line 42  
 ORA-04088: error during execution of trigger 'ANDITOADER.EX\_11'

```

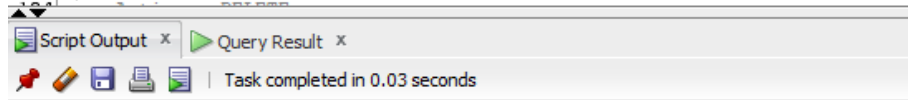
-- Activare UPDATE
-- STOC INDISPONIBIL
UPDATE contracte SET cantitate_prod = 100000 WHERE id_comanda = 100;
-- DISTRIBUITOR OCUPAT
UPDATE contracte SET cantitate_prod = 50000 WHERE id_comanda = 2;
-- Varianta acceptata
UPDATE contracte SET cantitate_prod = 2000 WHERE id_comanda = 1;
SELECT * FROM contracte;

```

```

115 -- Activare UPDATE
116 -- STOC INDISPONIBIL
117 UPDATE contracte SET cantitate_prod = 100000 WHERE id_comanda = 2;
118 -- DISTRIBUTOR OCUPAT
119 UPDATE contracte SET cantitate_prod = 50000 WHERE id_comanda = 1;
120 -- Varianta acceptata
121 UPDATE contracte SET cantitate_prod = 2000 WHERE id_comanda = 1;
122 SELECT * FROM contracte;
123

```



```

Error starting at line : 117 in command -
UPDATE contracte SET cantitate_prod = 100000 WHERE id_comanda = 2
Error report -
ORA-20110: STOC INDISPONIBIL!
ORA-06512: at "ANDITOADER.EX_11", line 40
ORA-04088: error during execution of trigger 'ANDITOADER.EX_11'

```

```

Error starting at line : 119 in command -
UPDATE contracte SET cantitate_prod = 50000 WHERE id_comanda = 1
Error report -
ORA-20111: DISTRIBUTOR OCUPAT!
ORA-06512: at "ANDITOADER.EX_11", line 42
ORA-04088: error during execution of trigger 'ANDITOADER.EX_11'

```

1 row updated.

```

-- Activare DELETE
DELETE FROM contracte WHERE id_comanda = 1;
-- Varianta acceptata
DELETE FROM contracte WHERE id_comanda = 2;
SELECT * FROM contracte;

```

```

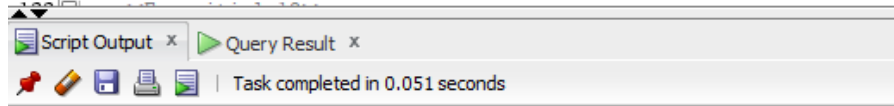
-- Stergere TRIGGER
DROP TRIGGER ex_11;

```

```

124 | -- Activare DELETE
125 | DELETE FROM contracte WHERE id_comanda = 1;
126 | -- Varianta acceptata
127 | DELETE FROM contracte WHERE id_comanda = 2;
128 | SELECT * FROM contracte;
129 |
130 | -- Stergere TRIGGER
131 | DROP TRIGGER ex_11;
132 |

```



```

ORA-20111: DISTRIBUITOR OCUPAT!
ORA-06512: at "ANDITOADER.EX_11", line 42
ORA-04088: error during execution of trigger 'ANDITOADER.EX_11'

```

1 row updated.

```

Error starting at line : 125 in command -
DELETE FROM contracte WHERE id_comanda = 1
Error report -
ORA-20112: CONTRACT IN DESFASURARE!
ORA-06512: at "ANDITOADER.EX_11", line 44
ORA-04088: error during execution of trigger 'ANDITOADER.EX_11'

```

1 row deleted.

## 6.7 Exercițiul 12

Definiți un trigger de tip LDD. Declanșați trigger-ul.

```

--      Definiti un trigger de tip LDD care sa permita modificarea schemei doar
de catre utilizatorul
-- anditoader. Salvati toate modificarile facute in tabela istoric_user.

```

```

CREATE TABLE istoric_admin (
    utilizator VARCHAR(30),
    nume_bd VARCHAR(50),
    eveniment VARCHAR(20),
    nume_obiect VARCHAR(30),
    data_ev DATE
);
/

```

```

CREATE OR REPLACE TRIGGER ex_12
    BEFORE CREATE OR DROP OR ALTER ON SCHEMA
BEGIN
    IF USER != UPPER('anditoader') THEN
        RAISE_APPLICATION_ERROR(-20120,'Doar administratorul poate aduce
schimbări bazei de date!');
    
```

```

        END IF;
        INSERT INTO istoric_admin VALUES (SYS.LOGIN_USER, SYS.DATABASE_NAME,
SYS.SYSEVENT, SYS.DICTIONARY_OBJ_NAME, SYSDATE);
END;
/
-- Verificare TRIGGER
ALTER TABLE date_angajati ADD calificare_necesara VARCHAR(30);
ALTER TABLE date_angajati DROP COLUMN calificare_necesara;
select * from istoric_admin;
ROLLBACK;
/
-- Stergere trigger
DROP TRIGGER ex_12;

139 CREATE TABLE istoric_admin (
140     utilizator VARCHAR(30),
141     nume_bd VARCHAR(50),
142     eveniment VARCHAR(20),
143     nume_obiect VARCHAR(30),
144     data_ev DATE
145 );
146 /
147
148 CREATE OR REPLACE TRIGGER ex_12
149     BEFORE CREATE OR DROP OR ALTER ON SCHEMA
150 BEGIN
151     IF USER != UPPER('anditoader') THEN
152         RAISE_APPLICATION_ERROR(-20120, 'Doar administratorul poate aduce schimbari bazei de date!');
153     END IF;
154     INSERT INTO istoric_admin VALUES (SYS.LOGIN_USER, SYS.DATABASE_NAME, SYS.SYSEVENT, SYS.DICTIONARY_OBJ_NAME, SYSDATE);
155 END;
156 /
157 -- Verificare TRIGGER
158 ALTER TABLE date_angajati ADD calificare_necesara VARCHAR(30);
159 ALTER TABLE date_angajati DROP COLUMN calificare_necesara;
160 select * from istoric_admin;
161 ROLLBACK;
162 /
163 -- Stergere trigger
164 DROP TRIGGER ex_12;

```

Script Output x Query Result x

All Rows Fetched: 4 in 0.001 seconds

	UTILIZATOR	NUME_BD	EVENTIMENT	NUME_OBIECT	DATA_EV
1	ANDITOADER	XE	ALTER	DATE_ANGAJATI	12-JAN-23
2	ANDITOADER	XE	ALTER	DATE_ANGAJATI	12-JAN-23
3	ANDITOADER	XE	ALTER	DATE_ANGAJATI	12-JAN-23
4	ANDITOADER	XE	ALTER	DATE_ANGAJATI	12-JAN-23

## 6.8 Exercițiul 13

Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.

SET serveroutput ON;

CREATE OR REPLACE PACKAGE pachet\_13 AS

    PROCEDURE ex\_6(v\_tip\_magazin IN lant\_companii.tip\_magazin%TYPE, v\_nume\_judet  
IN judete.nume\_judet%TYPE);

    PROCEDURE ex\_7(v\_tip\_magazin IN lant\_companii.tip\_magazin%TYPE);

    FUNCTION ex\_8(v\_nume\_companie IN lant\_companii.nume\_companie%TYPE) RETURN  
NUMBER;

    FUNCTION cel\_mai\_bun(v\_cod\_companie IN spatii\_comerciale.cod\_companie%TYPE,  
v\_cod\_localitate IN spatii\_comerciale.cod\_localitate%TYPE) RETURN



```

spatii_comerciale.cod_magazin%TYPE;
    PROCEDURE ex_9(v_nume_companie IN lant_companii.nume_companie%TYPE);
END pachet_13;
/
CREATE OR REPLACE PACKAGE BODY pachet_13 AS

-- <<Exercitiul 6>>

PROCEDURE ex_6 (v_tip_magazin IN lant_companii.tip_magazin%TYPE, v_nume_judet IN
judete.nume_jud%TYPE)
IS
    TYPE rec_magazin IS RECORD
        ( s_mag spatii_comerciale.suprafata_mag%TYPE,
          ca spatii_comerciale.cifra_vanzari%TYPE,
          localitate spatii_comerciale.cod_localitate%TYPE,
          companie spatii_comerciale.cod_companie%TYPE,
          an spatii_comerciale.data_deschidere%TYPE,
          cod spatii_comerciale.cod_magazin%TYPE
        );

    TYPE tab_magazine IS TABLE OF rec_magazin INDEX BY BINARY_INTEGER;
    t_mag_profil tab_magazine;

-- tabel imbricat
    TYPE tab_companii IS TABLE OF lant_companii.cod_companie%TYPE;
    t_comp_profil tab_companii;

    TYPE tab_localitati IS TABLE OF localitati.cod_localitate%TYPE;
    t_loc tab_localitati;

    v_cod_judet judete.cod_judet%TYPE;
    v_comp lant_companii.nume_companie%TYPE;
    v_loc localitati.nume_loc%TYPE;
    v_ca_totala lant_companii.cifra_afaceri%TYPE;

BEGIN
    BEGIN -- verificarea datelor introduse
        SELECT cod_judet
        INTO v_cod_judet
        FROM judete
        WHERE lower(nume_jud) = lower(v_nume_judet);
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            v_cod_judet := NULL;
            DBMS_OUTPUT.PUT_LINE (upper(v_nume_judet)||' nu este un judet!');
    END;
    IF v_cod_judet IS NOT NULL THEN
        SELECT cod_companie
        BULK COLLECT INTO t_comp_profil
        FROM lant_companii

```

```

WHERE lower(tip_magazin) = lower(v_tip_magazin);

IF t_comp_profil.COUNT = 0 THEN -- verificarea datelor introduse
    DBMS_OUTPUT.PUT_LINE ('Acest tip de magazin:
'||upper(v_tip_magazin)||', nu este inregistrat in baza de date!');
ELSE
    DBMS_OUTPUT.PUT_LINE ('<< Situatia magazinelor tip ' ||
upper(v_tip_magazin) || ' din judetul ' || upper(v_ume_judet) || '>>');
    SELECT cod_localitate
    BULK COLLECT INTO t_loc
    FROM localitati
    WHERE v_cod_judet = cod_judet;

    SELECT suprafata_mag, cifra_vanzari, cod_localitate, cod_companie,
data_deschidere, cod_magazin
    BULK COLLECT INTO t_mag_profil
    FROM spatii_comerciale
    ORDER BY cod_companie, cod_localitate, cifra_vanzari, suprafata_mag;

    FOR i IN t_mag_profil.FIRST..t_mag_profil.LAST LOOP
        IF t_mag_profil(i).companie MEMBER OF t_comp_profil AND
t_mag_profil(i).localitate MEMBER OF t_loc THEN

            SELECT nume_companie, cifra_afaceri
            INTO v_comp, v_ca_totala
            FROM lant_companii
            WHERE cod_companie = t_mag_profil(i).companie;

            SELECT nume_loc
            INTO v_loc
            FROM localitati
            WHERE cod_localitate = t_mag_profil(i).localitate;

            DBMS_OUTPUT.PUT_LINE('<O> ' || upper(v_comp) || ' ' ||
t_mag_profil(i).cod || ' ' || v_loc);
            DBMS_OUTPUT.PUT_LINE(' # Procent cira afaceri: ' ||
trunc(t_mag_profil(i).ca/ v_ca_totala, 3)|| '%');
            DBMS_OUTPUT.PUT_LINE(' # An deschidere: ' ||
t_mag_profil(i).an);
            DBMS_OUTPUT.PUT_LINE(' # Suprafata: ' ||
t_mag_profil(i).s_mag || ' m.p. ');
        END IF;
    END LOOP;

END IF;
END IF;
END ex_6;

-- <<Exercitiul 7>>

```

```
--      Sa se afiseze pentru magazinele de un anumit tip (discount, constructii,
proximitate) situatia actuala
-- a aprovizionarii. Se va calcula pretul aprovizionarii fiecarui magazin, stiind
ca marfa de la un centru logistic este impartita
-- in mod egal intre magazine.
```

```
PROCEDURE ex_7 (v_tip_magazin IN lant_companii.tip_magazin%TYPE) IS
    CURSOR magazine IS
        SELECT cod_magazin, lc.cod_companie, nume_companie, nume_loc, cod_logistic
        FROM spatii_comerciale sc, lant_companii lc, localitati l
        WHERE sc.cod_companie = lc.cod_companie
              AND sc.cod_localitate = l.cod_localitate
        ORDER BY nume_companie, l.nume_loc;

    CURSOR aprovizionare (centru_logistic contracte.cod_logistic%TYPE) IS
        SELECT nume_producator, nume_produc, tara, cantitate_prod,
nume_distribuitor, p.pret_unitar, c.id_comanda
        FROM contracte c, distribuitor d, producatori p
        WHERE c.cod_producator = p.cod_producator
              AND c.cod_distribuitor = d.cod_distribuitor
              AND c.cod_logistic = centru_logistic;

    v_cod_magazin spatii_comerciale.cod_magazin%TYPE;
    v_cod_companie spatii_comerciale.cod_companie%TYPE;
    v_nume_companie lant_companii.nume_companie%TYPE;
    v_nume_loc localitati.nume_loc%TYPE;
    v_cod_logistic spatii_comerciale.cod_logistic%TYPE;
    v_aux INTEGER;
    TYPE tab_mag_tip IS TABLE OF lant_companii.cod_companie%TYPE;
    mag_tip tab_mag_tip;

    BEGIN
        -- se inregistreaza companiile care sunt de tipul introdus
        SELECT cod_companie BULK COLLECT INTO mag_tip
        FROM lant_companii
        WHERE LOWER(v_tip_magazin) = LOWER(tip_magazin);
        -- DBMS_OUTPUT.PUT_LINE(mag_tip.COUNT);
        OPEN magazine;
        LOOP
            FETCH magazine INTO v_cod_magazin, v_cod_companie, v_nume_companie,
v_nume_loc, v_cod_logistic;
            EXIT WHEN magazine%NOTFOUND;
            -- DBMS_OUTPUT.PUT_LINE(v_cod_companie);
            -- pentru magazinele care sunt de tipul introdus se afiseaza
aprovizionarea
            IF v_cod_companie MEMBER OF mag_tip THEN
                DBMS_OUTPUT.PUT_LINE('<O> '||upper(v_nume_companie) || ' ' ||
v_cod_magazin || ' ' || v_nume_loc);
                FOR a IN aprovizionare (v_cod_logistic) LOOP
                    -- se calculeaza numarul de magazine care se aprovizioneaza de
```

```

la acelasi
        -- centru logistic
        SELECT COUNT(v_cod_magazin) INTO v_aux FROM spatii_comerciale
WHERE cod_logistic = v_cod_logistic;
        DBMS_OUTPUT.PUT_LINE (' ' || '<< Aprovizionare lunara
comanda' || a.id_comanda || '>>');
        DBMS_OUTPUT.PUT_LINE (' # ' || a.ume_produ || ', ' ||
ROUND(a.cantitate_prod/v_aux) || ' unitati, pret total ' ||
ROUND(a.cantitate_prod/v_aux*a.pret_unitar) || ' RON');
        DBMS_OUTPUT.PUT_LINE (' # Detalii:');
        DBMS_OUTPUT.PUT_LINE(' - Producator: ' ||
a.ume_producator || ', ' || a.tara);
        DBMS_OUTPUT.PUT_LINE(' - Distribuitor: ' ||
a.ume_distribuitor);
        END LOOP;
    END IF;
END LOOP;
CLOSE magazine;
END ex_7;

-- <<Exercitiul 8>>

--      Sa se afiseze pentru o companie x pretul mediu platit per minut pentru
publicitate televizata.

FUNCTION ex_8(v_ume_companie IN lant_companii.ume_companie%TYPE)
RETURN NUMBER IS v_pret_mediu FLOAT := NULL;
    v_cod_companie lant_companii.cod_companie%TYPE;
    exc_fara_televizat EXCEPTION;
BEGIN
    SELECT cod_companie INTO v_cod_companie
    FROM lant_companii
    WHERE lower(v_ume_companie) = lower(ume_companie);

    -- calculez pretul mediu in functie de toate contractele televizate aflate
in desfasurare
    BEGIN
        SELECT ROUND(SUM(t.difuzare_int1*60/t.durata_spot*pt.cost_int1 +
t.difuzare_int2*60/t.durata_spot*pt.cost_int2) /
SUM(t.difuzare_int1 + t.difuzare_int2))
        INTO v_pret_mediu
        FROM televizat t, post_tv pt, campanii_marketing cm
        WHERE t.cod_tv = pt.cod_tv
            AND t.cod_marketing = cm.cod_marketing
            AND cm.cod_companie = v_cod_companie
            AND cm.finalizare - SYSDATE >0
        GROUP BY cm.cod_companie;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            RAISE_APPLICATION_ERROR(-20080, 'Nu exista contracte televizate in
desfasurare!');

```

```

END;
RETURN v_pret_mediu;
EXCEPTION
WHEN NO_DATA_FOUND THEN
    RAISE_APPLICATION_ERROR(-20081, 'Nu exista magazinul introdus!');
END ex_8;

-- <<Exercitiul 9>>

--      Sa se afiseze pentru cel mai performant magazin X din fiecare localitate
--      (cea mai mare cifra de afaceri) informatii
--      legate de campaniile offline desfasurate de companie.

FUNCTION cel_mai_bun(v_cod_companie IN spatii_comerciale.cod_companie%TYPE,
v_cod_localitate IN spatii_comerciale.cod_localitate%TYPE)
RETURN spatii_comerciale.cod_magazin%TYPE IS
v_cod spatii_comerciale.cod_magazin%TYPE := -1;
BEGIN
    SELECT cod_magazin INTO v_cod
    FROM spatii_comerciale
    WHERE cifra_vanzari =(
                                SELECT MAX(cifra_vanzari)
                                FROM spatii_comerciale
                                WHERE cod_companie = v_cod_companie
                                AND cod_localitate = v_cod_localitate);

    RETURN v_cod;
END cel_mai_bun;

PROCEDURE ex_9
(v_num_companie IN lant_companii.num_companie%TYPE)
IS
    v_cod_companie lant_companii.cod_companie%TYPE;
    v_cod_loc localitati.cod_localitate%TYPE;
    v_cod_mag_1 spatii_comerciale.cod_magazin%TYPE;
    v_ca spatii_comerciale.cifra_vanzari%TYPE;
    TYPE rec_marketing IS RECORD(
        nume campanii_marketing.nume_companie%TYPE,
        nr_rev localitati_tinta.nr_reviste%TYPE,
        nr_pli localitati_tinta.nr_pliante%TYPE,
        nr_pan localitati_tinta.nr_panouri%TYPE
    );
    TYPE tab_marketing IS TABLE OF rec_marketing INDEX BY PLS_INTEGER;
    info_marketing tab_marketing;
-- selectez localitatile care au magazin(e) de la compania introdusa
CURSOR loc_mag_comp (v_cod_companie spatii_comerciale.cod_companie%TYPE) IS
    SELECT DISTINCT sc.cod_localitate, l.nume_loc, j.nume_jud
    FROM spatii_comerciale sc, localitati l, judete j
    WHERE cod_companie = v_cod_companie
        AND sc.cod_localitate = l.cod_localitate
        AND l.cod_judet = j.cod_judet
    ORDER BY j.nume_jud, l.nume_loc;

```

```

-- selectez detalii despre campaniile de marketing desfasurate de compania X
in toate localitatile
CURSOR info_campanii (v_cod_companie spatii_comerciale.cod_companie%TYPE) IS
    SELECT lt.cod_localitate, cm.nume_companie, nvl(lt.nr_reviste, 0) as rev,
nvl(lt.nr_pliante, 0) as pli, nvl(lt.nr_panouri, 0) as pan
    FROM localitati_tinta lt, camp_offline co, campanii_marketing cm
    WHERE lt.cod_offline = co.cod_offline
    AND co.cod_marketing = cm.cod_marketing
    AND cm.cod_companie = v_cod_companie
    AND cm.finalizare - SYSDATE > 0;
BEGIN
    SELECT cod_companie INTO v_cod_companie
    FROM lant_companii
    WHERE lower(v_nume_companie) = lower(nume_companie);
    -- salvez informatiile despre campaniile aflate in desfasurare de compani X in
toate localitatile
    FOR mkt IN info_campanii(v_cod_companie) LOOP
        info_marketing(mkt.cod_localitate).nume := mkt.nume_companie;
        info_marketing(mkt.cod_localitate).nr_rev := mkt.rev;
        info_marketing(mkt.cod_localitate).nr_pli := mkt.pli;
        info_marketing(mkt.cod_localitate).nr_pan := mkt.pan;
    END LOOP;

    FOR loc IN loc_mag_comp(v_cod_companie) LOOP
        v_cod_mag_1 := cel_mai_bun(v_cod_companie, loc.cod_localitate);
        SELECT cifra_vanzari INTO v_ca FROM spatii_comerciale WHERE cod_magazin =
v_cod_mag_1;
        DBMS_OUTPUT.PUT_LINE('<O>' || ' ' || upper(v_nume_companie) || ' ' ||
v_cod_mag_1 || ' ' || loc.nume_jud || ' ' || loc.nume_loc);
        DBMS_OUTPUT.PUT_LINE(' # Cifra vanzari: ' || v_ca || ' RON/luna');
        -- afisez info despre campanii de marketing, daca ea exista
        BEGIN
            DBMS_OUTPUT.PUT_LINE(' # Campania:
' || upper(info_marketing(loc.cod_localitate).nume));
            DBMS_OUTPUT.PUT_LINE(' - Numar reviste: ' ||
info_marketing(loc.cod_localitate).nr_rev);
            DBMS_OUTPUT.PUT_LINE(' - Numar pliante: ' ||
info_marketing(loc.cod_localitate).nr_pli);
            DBMS_OUTPUT.PUT_LINE(' - Numar panouri publicitare: ' ||
info_marketing(loc.cod_localitate).nr_pan);
        EXCEPTION
            WHEN NO_DATA_FOUND THEN
                DBMS_OUTPUT.PUT_LINE(' # NU EXISTA CAMPANIE DE MARKETING AFLATA IN
DESFASURARE! ');
        END;
    END LOOP;

    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            RAISE_APPLICATION_ERROR(-20090, 'Magazinul introdus nu a fost
gasit!');

```

```

        WHEN TOO_MANY_ROWS THEN
            RAISE_APPLICATION_ERROR(-20091, 'Exista mai multe magazine cu aceeasi
cifra de vanzari!');
        WHEN CURSOR_ALREADY_OPEN THEN
            RAISE_APPLICATION_ERROR(-20092, 'Accesare multipla a aceeasi zona de
memorie!');
        WHEN OTHERS THEN
            RAISE_APPLICATION_ERROR(-20093, 'Alta eroare!');
END ex_9;

END pachet_13;

-- <<Exercitiul 6>>
execute pachet_13.ex_6('discount', 'dâmbovita');
-- <<Exercitiul 7>>
execute pachet_13.ex_7('discount');
-- <<Exercitiul 8>>
BEGIN
    DBMS_OUTPUT.PUT_LINE(pachet_13.ex_8('kaufland') || ' RON/minut');
END;
/
-- <<Exercitiul 9>>
execute pachet_13.ex_9('kaufland');

```

## 6.8 Exercițiul 14

Definiți un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri).

```

-- <<Exercitiul 14>>
--    In luna ianuarie a fiecarui an, magazinele isi actualizeaza cifra lunara de vanzari, iar
produsorii si posturile TV tarifele.
--    Sa se actualizeze baza de date in functie de noile valori si sa se afiseze schimbarile.
SET serveroutput ON;

CREATE OR REPLACE PACKAGE pachet_14 AS
    TYPE rec_cmp IS RECORD
    (
        cod_lant_companii.cod_companie%TYPE,
        nume_lant_companii.nume_companie%TYPE,
        ca_v_lant_companii.cifra_afaceri%TYPE,
        ca_n_lant_companii.cifra_afaceri%TYPE
    );
    TYPE rec_mkt IS RECORD
    ( --cod_companie campanii_marketing.cod_companie%TYPE,

```

```

        cod_marketing campanii_marketing.cod_marketing%TYPE,
        suma NUMBER(12,2)
    );
    TYPE tab_situatie_mkt IS TABLE OF rec_mkt;
    t_situatie_mkt tab_situatie_mkt;

    TYPE tab_situatie_companii IS TABLE OF rec_cmp INDEX BY PLS_INTEGER;
    t_situatie_companii tab_situatie_companii;

    FUNCTION actualizare_cifra_afaceri(v_cod_companie IN
    lant_companii.cifra_afaceri%TYPE) RETURN spatii_comerciale.cifra_vanzari%TYPE;
    PROCEDURE actualizare_companii;
    PROCEDURE actualizare;
    PROCEDURE actualizare_cmp_mkt;

END pachet_14;

CREATE OR REPLACE PACKAGE BODY pachet_14 AS
    FUNCTION actualizare_cifra_afaceri(v_cod_companie IN
    lant_companii.cifra_afaceri%TYPE)
    RETURN spatii_comerciale.cifra_vanzari%TYPE IS v_ca_noua
    spatii_comerciale.cifra_vanzari%TYPE;
    BEGIN
        SELECT SUM(cifra_vanzari) INTO v_ca_noua
        FROM spatii_comerciale sc, lant_companii lc
        WHERE sc.cod_companie = lc.cod_companie
        AND sc.cod_companie = v_cod_companie
        GROUP BY sc.cod_companie;
        RETURN v_ca_noua;
    END actualizare_cifra_afaceri;

    PROCEDURE actualizare_companii IS
    CURSOR c IS
        SELECT cod_companie, nume_companie
        FROM lant_companii
        ORDER BY cod_companie;
    v_ca_veche lant_companii.cifra_afaceri%TYPE;
    v_ca_noua lant_companii.cifra_afaceri%TYPE;
    BEGIN
        FOR companie IN c LOOP
            SELECT cifra_afaceri INTO v_ca_veche FROM lant_companii WHERE
            cod_companie = companie.cod_companie;
            v_ca_noua := actualizare_cifra_afaceri(companie.cod_companie);
            t_situatie_companii(companie.cod_companie).cod :=
            companie.cod_companie;
            t_situatie_companii(companie.cod_companie).nume :=
            companie.nume_companie;
            t_situatie_companii(companie.cod_companie).ca_v := v_ca_veche;
            t_situatie_companii(companie.cod_companie).ca_n := v_ca_noua;
            -- UPDATE lant_companii SET cifra_afaceri = v_ca_noua WHERE
            cod_companie = companie.cod_companie;
        END LOOP;
    END actualizare_companii;
END pachet_14;

```



```

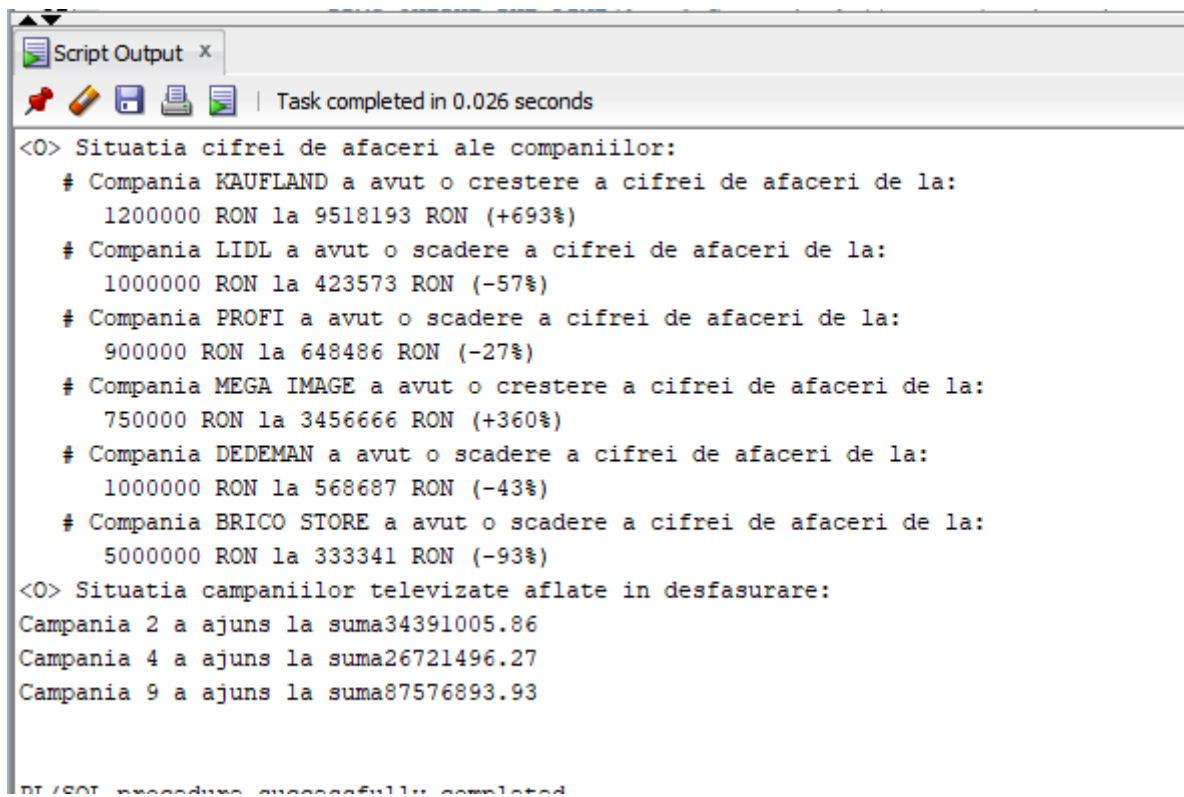
        END LOOP;
    END actualizare_companii;

    PROCEDURE actualizare_cmp_mkt IS
    BEGIN
        SELECT cm.cod_marketing,
        ROUND(SUM(t.difuzare_int1*60/t.durata_spot*pt.cost_int1 +
        t.difuzare_int2*60/t.durata_spot*pt.cost_int2)) * MIN(cm.finalizare - SYSDATE)
        suma
        BULK COLLECT INTO t_situatie_mkt
        FROM televizat t, post_tv pt, campanii_marketing cm
        WHERE t.cod_tv = pt.cod_tv
            AND t.cod_marketing = cm.cod_marketing
            AND cm.finalizare - SYSDATE > 0
        GROUP BY cm.cod_marketing;
    END actualizare_cmp_mkt;

    PROCEDURE actualizare IS
    v_aux lant_companii.cifra_afaceri%TYPE;
    BEGIN
        actualizare_companii;
        actualizare_cmp_mkt;
        DBMS_OUTPUT.PUT_LINE('<O> Situatia cifrei de afaceri ale companiilor:');
        FOR i IN t_situatie_companii.FIRST..t_situatie_companii.LAST LOOP
            v_aux :=
            (TRUNC(t_situatie_companii(i).ca_n/t_situatie_companii(i).ca_v-1, 2))*100;
            IF t_situatie_companii(i).ca_n - t_situatie_companii(i).ca_v > 0
            THEN
                DBMS_OUTPUT.PUT_LINE('    # Compania ' ||
                upper(t_situatie_companii(i).nume) || ' a avut o crestere a cifrei de afaceri de
                la:');
                DBMS_OUTPUT.PUT_LINE('        ' || t_situatie_companii(i).ca_v ||
                ' RON la ' || t_situatie_companii(i).ca_n || ' RON (+ ' || v_aux || '%)');
            ELSE
                DBMS_OUTPUT.PUT_LINE('    # Compania ' ||
                upper(t_situatie_companii(i).nume) || ' a avut o scadere a cifrei de afaceri de
                la:');
                DBMS_OUTPUT.PUT_LINE('        ' || t_situatie_companii(i).ca_v ||
                ' RON la ' || t_situatie_companii(i).ca_n || ' RON ( ' || v_aux || '%)');
            END IF;
        END LOOP;
        DBMS_OUTPUT.PUT_LINE('<O> Situatia campaniilor televizate aflate in
        desfasurare:');
        FOR i IN 1..t_situatie_mkt.COUNT LOOP
            DBMS_OUTPUT.PUT_LINE('Campania ' ||
            t_situatie_mkt(i).cod_marketing || ' a ajuns la suma ' ||
            t_situatie_mkt(i).suma);
        END LOOP;
    END actualizare;
END pachet_14;

```

```
execute pachet_14.actualizare;
```



```
<O> Situatia cifrei de afaceri ale companiilor:
# Compania KAUF LAND a avut o crestere a cifrei de afaceri de la:
  1200000 RON la 9518193 RON (+693%)
# Compania LIDL a avut o scadere a cifrei de afaceri de la:
  1000000 RON la 423573 RON (-57%)
# Compania PROFI a avut o scadere a cifrei de afaceri de la:
  900000 RON la 648486 RON (-27%)
# Compania MEGA IMAGE a avut o crestere a cifrei de afaceri de la:
  750000 RON la 3456666 RON (+360%)
# Compania DEDEMAN a avut o scadere a cifrei de afaceri de la:
  1000000 RON la 568687 RON (-43%)
# Compania BRICO STORE a avut o scadere a cifrei de afaceri de la:
  5000000 RON la 333341 RON (-93%)
<O> Situatia campaniilor televizate aflate in desfasurare:
Campania 2 a ajuns la suma34391005.86
Campania 4 a ajuns la suma26721496.27
Campania 9 a ajuns la suma87576893.93

BT /SQL procedure successfully completed
```